

Development of a Three Dimensional Plasma Fluid Code Using a Lagrange-Monte-Carlo Scheme

Ryoko TATSUMI, Alexei RUNOV¹⁾, Ralf SCHNEIDER²⁾, Kazuo HOSHINO
and Akiyoshi HATAYAMA

Faculty of Science and Technology, Keio University, Yokohama 223-8522, Japan

¹⁾*Max-Planck-Institute for Plasma Physics, Greifswald, Germany*

²⁾*University of Greifswald, Greifswald, Germany*

(Received 28 August 2019 / Accepted 17 November 2019)

We are developing a Lagrange (LG)-Monte-Carlo (MC) scheme for three-dimensional (3D) SOL/Divertor plasma fluid modeling. By using test particles, the scheme is suitable for handling 3D complex geometries. The semi-implicit treatment of the pressure gradient term enables us to improve the robustness of the coupling of the continuity and the momentum equations. Detailed numerical checks of the integrated scheme of LG-MC have been done for a simple 1D geometry. Benchmark tests between the new LG-MC and a conventional Finite-Volume scheme were carried out and good agreement was obtained. A first test calculation for a 3D cylindrical geometry has been also successfully done.

© 2020 The Japan Society of Plasma Science and Nuclear Fusion Research

Keywords: SOL/Divertor plasma, fluid simulation, Lagrange scheme

DOI: 10.1585/pfr.15.1403003

1. Introduction

To understand plasma transport in the SOL/Divertor region is one of the most significant challenges for fusion reactors. Aiming at that, several numerical studies have been done with plasma fluid codes, such as B2 [1–3], B2.5 [4], SOLDOR [5], EDGE2D [6], LINDA [7] etc. In most of the studies, however, the fluid models are in two-dimension (2D), assuming axial symmetry of devices. Since the symmetry does not hold even in tokamaks due to 1) implementation of resonant magnetic perturbation (RMP) field and 2) presence of some non-axisymmetric structures such as gas pumping/puffing ports, there has been a great demand of a three-dimensional (3D) plasma fluid code.

One scheme to handle 3D complex geometries is the so-called Monte-Carlo scheme (MC) which uses “fluid pseudo particles” to solve the transport equations. This is used for example in E3D [8,9] and EMC3 [10,11]. We also developed a Monte-Carlo code to solve the transport equations [12] and obtained correct solutions for diffusive-dominant cases. To extend the scheme to convective-dominant cases, we started developing a semi-Lagrange scheme (LG) [13]. Due to its semi-implicit treatment of the pressure gradient term, LG turned out to be more suitable for the convective-dominant problem.

In LG, diffusive parts are treated by updating weights of the pseudo particles, which will be explained in Sec. 3.3. However, treating them by random walk steps, i.e., with MC, has an advantage in complex magnetic configuration,

like configurations with ergodic magnetic fields. Therefore, we developed the Lagrange-Monte-Carlo scheme (LG-MC), adding diffusive parts treated by MC coupled with LG. To validate numerical algorithms of LG and LG-MC, a 1D benchmark test for the three schemes, a Finite Volume scheme (FV), LG, and LG-MC, was done.

After confirming the validity of 1D LG and LG-MC through the benchmark, the LG-MC code was extended to a 3D cylindrical geometry. This paper is constructed as follows. In Sec. 2, 1D basic equations along the field line are given. In Sec. 3, starting with LG, we give a description of numerical algorithm for LG-MC. Results of the 1D benchmark test will be presented in Sec. 4. Finally we introduce the result of the 3D LG-MC code in Sec. 5 and conclude the paper in Sec. 6.

2. Basic Equations (1D)

Assuming a typical SOL/divertor geometry of a tokamak, the spatial coordinate of the 1D model, x , is taken along the magnetic field from the stagnation point ($x = 0$) to the divertor plate ($x = L_x$) (See Fig. 1). Four variables, the plasma density n , the fluid velocity along the magnetic field V , the ion and the electron temperatures, T_i and T_e , are solved by the set of plasma fluid equations for hydrogen ions (H^+) and electrons, originally derived by Braginskii [14]:

$$\frac{\partial n}{\partial t} + \frac{\partial}{\partial x} (nV) = S_n = S_{input} + mn_n \langle \sigma v \rangle_{ion}. \quad (1)$$

$$\frac{\partial}{\partial t} (mnV) + \frac{\partial}{\partial x} \left(mnV^2 + p - \frac{4}{3} \eta_i \frac{\partial V}{\partial x} \right) = S_{m,coll}. \quad (2)$$

author's e-mail: tatsumi@ppl.appi.keio.ac.jp

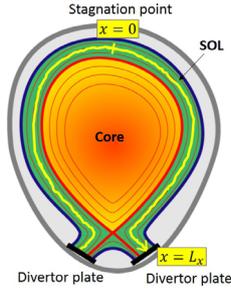


Fig. 1 Schematic view of the 1D SOL model.

$$\frac{\partial}{\partial t} \left(\frac{3}{2} p_i \right) + \frac{\partial}{\partial x} \left(\frac{3}{2} p_i V - \kappa_i \frac{\partial T_i}{\partial x} \right) = S_{T_i}, \quad p_i = n T_i, \quad (3)$$

$$S_{T_i} = Q_{input} - p_i \frac{\partial V}{\partial x} + k_{th}(T_e - T_i) + \frac{4}{3} \eta_i \left(\frac{\partial V}{\partial x} \right)^2 + \frac{m}{2} V^2 S_n - V S_{m,coll}.$$

$$\frac{\partial}{\partial t} \left(\frac{3}{2} p_e \right) + \frac{\partial}{\partial x} \left(\frac{3}{2} p_e V - \kappa_e \frac{\partial T_e}{\partial x} \right) = S_{T_e}, \quad p_e = n T_e, \quad (4)$$

$$S_{T_e} = Q_{input} - p_e \frac{\partial V}{\partial x} - k_{th}(T_e - T_i) + E_{e,ion} n n_n \langle \sigma v \rangle_{ion}.$$

The momentum equation is the sum of those of the ions and the electrons, with the assumption of small mass ratio ($m_i \gg m_e$). Thus, m is the ion mass m_i , the viscosity coefficient is calculated as a production of the ion viscosity coefficient η_{0i} and the ion temperature, $\eta_i = \eta_{0i} T_i^{5/2}$, and $p = p_i + p_e = n(T_i + T_e)$. In Eqs. (3) and (4), κ_s ($= \kappa_{0s} T_s^{5/2}$, $s = i, e$) is the heat conductivity, where κ_{0s} is a constant heat conductivity coefficient. The symbol k_{th} is the coefficient of energy-exchange between the electrons and the ions. As a collision process, only the ionization is taken into account in our model at present. The symbols $\langle \sigma v \rangle_{ion}$ and $E_{e,ion}$ represent the rate coefficient and the electron energy loss by ionization, respectively.

The neutral density n_n is modeled by

$$\frac{\partial n_n}{\partial t} + \frac{\partial}{\partial x} (n_n V_n) = -n_n n \langle \sigma v \rangle_{ion}, \quad (5)$$

where V_n is the neutral velocity towards the upstream of the plasma. By assuming V_n to be a constant, the steady state solution is given as

$$n_n(x) = n_n(L_x) \exp \left[\int_x^{L_x} \frac{n \langle \sigma v \rangle_{ion}}{V_n} dx \right], \quad (6)$$

$$V_n < 0, n_n(L_x) = R \frac{n(L_x) V(L_x)}{V_n},$$

where R is the recycling coefficient.

3. Lagrange-Monte-Carlo Scheme

In the Lagrange-Monte-Carlo Scheme (LG-MC), we introduce test particles representing a plasma quantity in the j -th cell, f_j , as

$$\frac{\sum_{p=1}^{N_{f,j}} W_f(p)}{\Delta V_j} = f_j,$$

$$f = n, M(\equiv mnV), \epsilon_{i,e} \left(\equiv \frac{3}{2} p_{i,e} \right), \quad (7)$$

where $N_{f,j}$, $W_f(p)$, and ΔV_j are the total number of test particles in the j -th spatial cell, the weight for f of the p -th particle, and the volume of the j -th cell, respectively. In order to solve the four physical quantities as f in the respective transport equations, four different species of those particles are prepared. For calculating the pressure gradient, we use staggered grids, i.e., alternating mesh for the density and the momentum/velocity, to avoid numerical instabilities.

As mentioned in Sec. 1, the difference between LG and LG-MC is whether the diffusive parts are treated by changing the weights of the test particles or by random walk steps, i.e., with a Monte-Carlo scheme (MC). In this section, after describing the scheme of LG (Sec. 3.1 - 3.4) and MC (Sec. 3.5 - 3.6) separately, we will explain the integrated scheme, LG-MC, in Sec. 3.7.

3.1 Basic equations for LG

In LG, Eqs. (1) - (4) are rewritten with the Lagrange derivative, $\frac{d}{dt} \equiv \frac{\partial}{\partial t} + V \frac{\partial}{\partial x}$, as

$$\frac{dn}{dt} = -n \frac{\partial V}{\partial x} + S_n. \quad (8)$$

$$\frac{d}{dt} (mnV) = -mnV \frac{\partial V}{\partial x} - \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(\frac{4}{3} \eta_i \frac{\partial V}{\partial x} \right) + S_{m,coll}. \quad (9)$$

$$\frac{d}{dt} \left(\frac{3}{2} p_i \right) = -\frac{3}{2} p_i \frac{\partial V}{\partial x} + S_{T_i} + \frac{\partial}{\partial x} \left(\kappa_i \frac{\partial T_i}{\partial x} \right). \quad (10)$$

$$\frac{d}{dt} \left(\frac{3}{2} p_e \right) = -\frac{3}{2} p_e \frac{\partial V}{\partial x} + S_{T_e} + \frac{\partial}{\partial x} \left(\kappa_e \frac{\partial T_e}{\partial x} \right). \quad (11)$$

3.2 Transport of the particles in LG

To treat the convective transport semi-implicitly, Eqs. (8) and (9) are combined to obtain

$$mn \frac{dV}{dt} = -mVS_n - \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(\frac{4}{3} \eta_i \frac{\partial V}{\partial x} \right) + S_{m,coll}. \quad (12)$$

From Eq. (12), the “new” velocity is obtained at each staggered mesh as

$$V^{new} = V + \frac{\Delta t}{mn} \left(-mVS_n - \frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(\frac{4}{3} \eta_i \frac{\partial V}{\partial x} \right) + S_{m,coll} \right). \quad (13)$$

The convective transport of the p -th particle from the old position $x^{old}(p)$ to the new position $x^{new}(p)$ is described as

$$x^{new}(p) = x^{old}(p) + V_\mu(p) \Delta t, \quad (14)$$

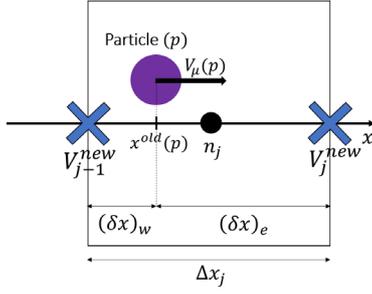


Fig. 2 Transport of the p -th particle with the interpolated velocity $V_\mu(p)$.

where $V_\mu(p)$ is a linearly interpolated V^{new} at $x^{old}(p)$,

$$V_\mu(p) = \frac{(\delta x)_e}{\Delta x_j} V_{j-1}^{new} + \frac{(\delta x)_w}{\Delta x_j} V_j^{new}. \quad (15)$$

as shown in Fig. 2.

3.3 Injection and weight-change of the particles

The first terms on the right-hand-side (RHS) of Eqs. (8) - (11) represent compression/expansion of each quantity f . This effect is automatically included when we push each particle with the velocity V_μ (See Eq. (15)) and count the number of those particles in Eulerian cells (See mathematical description in Appendix). Hence, only the rest of terms should be treated either by 1) adding new particles or by 2) changing the particles' weight, W_f . At present, only the particle source term S_n is treated by adding new particles, while the other source/sink terms in the momentum/energy equations are treated by weight-change.

For 1), the number of particles added in the j -th cell from the source term $S_{n,j}$ is calculated as

$$N_{add,j} = \frac{S_{n,j} \Delta V_j \Delta t}{W_n}. \quad (16)$$

At each time step Δt , $N_{add,j}$ test particles are uniformly added in each cell.

For 2), taking the momentum equation Eq. (9) as an example, the weight-change should be described as

$$\frac{M^{new} - M}{\Delta t} = -\frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(\frac{4}{3} \eta_i \frac{\partial V}{\partial x} \right) + S_{m,coll}. \quad (17)$$

It should be noted again that the term $-M \frac{\partial V}{\partial x}$ on RHS has been eliminated from Eq. (9), because the effect of this term is already treated by the convective transport. From Eq. (17), M^{new} is obtained as

$$\begin{aligned} M^{new} &= \frac{W_M^{new} N_{s,j}}{\Delta V_{s,j}} \\ &= M + \left[-\frac{\partial p}{\partial x} + \frac{\partial}{\partial x} \left(\frac{4}{3} \eta_i \frac{\partial V}{\partial x} \right) + S_{m,coll} \right] \Delta t. \end{aligned} \quad (18)$$

The new momentum density $M_{s,j}^{new}$ should be represented by the LG particles in the j -th staggered cell. To satisfy this, W_M of all the LG particles in the j -th staggered cell are replaced as

$$W_M^{new} = \frac{M^{new} \Delta V_{s,j}}{N_{s,j}}, \quad (19)$$

where $N_{s,j}$ is the total number of the particles in the j -th staggered cell after the source particles are added. Similarly, the weights for the internal energy $W_{\epsilon_{i/e}}$ are updated using the energy equations.

3.4 Setting of boundary conditions in LG

At $x = L_x$, the following conditions are set.

$$V(L_x) \geq C_s(L_x) = \sqrt{\frac{T_i(L_x) + T_e(L_x)}{m}}, \quad (20)$$

for the velocity $V(L_x)$, and

$$\frac{5}{2} n V T_i + \frac{1}{2} m n V^3 - \frac{4}{3} \eta_i V \frac{\partial V}{\partial x} - \kappa_i \frac{\partial T_i}{\partial x} = \gamma_i n V T_i, \quad (21)$$

$$\frac{5}{2} n V T_e - \kappa_e \frac{\partial T_e}{\partial x} = \gamma_e n V T_e, \quad (22)$$

for the ion/electron total heat fluxes.

To set the condition (22) by LG, we reformulate it as

$$q_{econd,req} \equiv -\kappa_e \frac{\partial T_e}{\partial x} \Big|_{x=L_x} = \gamma_e n V T_e - \frac{5}{2} n V T_e, \quad (23)$$

and replace the third term on RHS of Eq. (11) as

$$\frac{\partial}{\partial x} \left(\kappa_e \frac{\partial T_e}{\partial x} \right) = -\frac{q_{econd,req} - \left(-\kappa_e \frac{\partial T_e}{\partial x} \right)}{\Delta x}, \quad (24)$$

when the weight W_{ϵ_e} of the test particles in the last cell are updated with Eq. (11). The same procedure is done for the ion heat flux with the required value,

$$q_{icond,req} = \gamma_i n V T_i - \frac{5}{2} n V T_i - \frac{1}{2} m n V^3 + \frac{4}{3} \eta_i V \frac{\partial V}{\partial x}. \quad (25)$$

3.5 Monte-Carlo scheme (MC)

The Monte-Carlo scheme (MC) is used for treating diffusive terms by random walk processes. We consider a stochastic quantity $x(t)$ obeying an Ito stochastic differential equation (SDE) written as [15]

$$dx(t) = a[x(t), t] dt + b[x(t), t] dW(t), \quad (26)$$

where $dW(t)$ is an increment of a Wiener process, defined by

$$dW(t) = W(t + dt) - W(t), \quad (27)$$

which satisfies the mean value $\langle dW(t) \rangle = 0$ and the variance $\langle dW(t)^2 \rangle = dt$. In that case, it can be derived that a

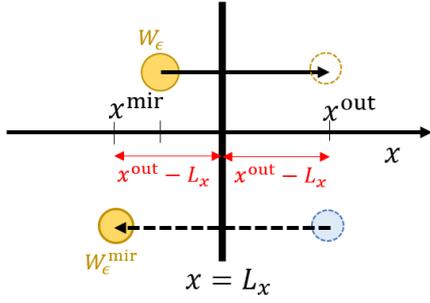


Fig. 3 Schematic view of the flux boundary condition in MC.

probability density $p(x, t)$ obeys the Fokker-Planck equation (FPE),

$$\frac{\partial p(x, t)}{\partial t} = -\frac{\partial}{\partial x} [a(x, t)p(x, t)] + \frac{1}{2} \frac{\partial^2}{\partial x^2} [b^2(x, t)p(x, t)]. \quad (28)$$

If we take the stochastic quantity $x(t)$ as a particle position, Eq. (26) denotes a transport process with a drift coefficient $a(x, t)$ and a diffusion coefficient $\frac{1}{2}b^2(x, t)$. Then a probability finding a particle at a position x at a time t directly corresponds to the probability density $p(x, t)$. By counting particles transported by Eq. (26), we obtain an expected profile of $p(x, t)$ in each cell.

In this manner, we calculate the viscosity part (the third term on RHS of Eq. (9)) by the random walk of the particles having the weight W_M , and the conductive part (the third terms on RHS of Eq. (10) and (11)) by that of the particles having the weight W_{e_i} or W_{e_e} .

3.6 Boundary condition in MC

We set the conductive flux $q_{cond,req}$ (See Eqs. (23) and (25)) at the boundary ($x = L_x$) in MC. Since the procedure is the same for the ion conductive flux and that of electrons, here we omit the subscriptions i or e . Every time step, we compute the energy outflux $q_{cond,comp}$,

$$q_{cond,comp} = \frac{\sum_{p=1}^{N_{cross}} W_{\epsilon}(p)}{\Delta t}, \quad (29)$$

where N_{cross} is the number of particles that crossed the boundary in Δt . First, we nullify the outflux by reflecting those particles to the domain. Since each of those particles should keep the spatial length of the step, we perform mirror-reflection, calculating the reflected position x^{mir} as

$$x^{mir} = L_x - (x^{out} - L_x) = 2L_x - x^{out}, \quad (30)$$

where x^{out} is the position after crossing the boundary ($x^{out} > L_x$) (See Fig. 3).

If the particle has a different weight after being reflected,

$$W_{\epsilon}^{mir}(p) \equiv kW_{\epsilon}(p), \quad 0 < k < 1, \quad (31)$$

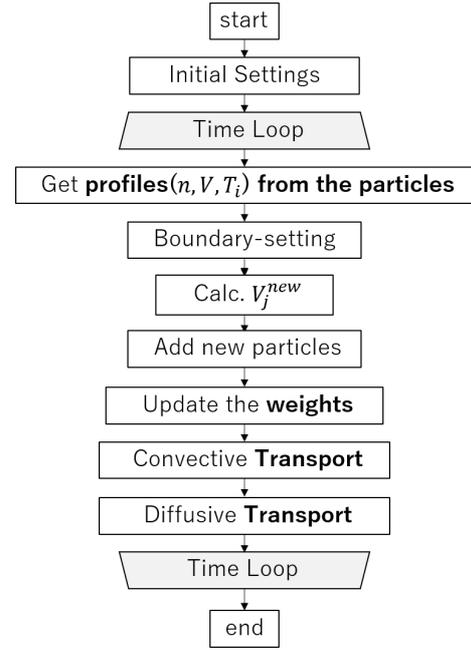


Fig. 4 Flow Chart of LG-MC.

the net flux q becomes non-zero, such as

$$q = \frac{\sum_{p=1}^{N_{cross}} W_{\epsilon}(p) - W_{\epsilon}^{mir}(p)}{\Delta t} = (1 - k)q_{cond,comp}. \quad (32)$$

The weight $W_{\epsilon}^{mir}(p)$ to set $q = q_{cond,req}$ is

$$W_{\epsilon}^{mir}(p) = kW_{\epsilon}(p) = \left(1 - \frac{q_{cond,req}}{q_{cond,comp}}\right) W_{\epsilon}(p). \quad (33)$$

This is the procedure to set the boundary condition, the conductive flux $q_{cond,req}$ at $x = L$. However, in the case $q_{cond,comp} < q_{cond,req}$, instead of performing the weight-change by Eq. (33), we simply absorb all the N_{cross} particles. This is the way to make $q_{cond,comp}$ maximum without introducing unphysical negative weight of the particles. By this transient treatment, $q_{cond,comp}$ gets larger, and the reflection treatment with Eq. (33) is performed again when it becomes larger than $q_{cond,req}$.

3.7 Integration of LG and MC

In LG, as described in Sec. 3.3, the diffusive terms are treated by changing the weights of the test particles (See Eq. (17) for the viscosity term). On the other hand, we have shown in Sec. 3.5 that they can also be treated by MC, which leads to the LG-MC scheme.

As shown in Fig. 4, we simply add the random walk process after the convective transport in LG-MC. The time step is determined by the convective transport in LG while it is restricted by the heat conduction in MC. This means the required time step of MC is smaller than that of LG in most of cases. Therefore, several iterations are done in MC

after one step of LG, so that it satisfies

$$\Delta t_{LG} = N_{MC} \delta t_{MC}, \quad (34)$$

where Δt_{LG} , δt_{MC} , and N_{MC} are the time step of LG and MC, and the number of iterations in MC, respectively.

4. A 1D Benchmark Test

To validate LG and LG-MC, a benchmark test was done with three schemes (FV, LG, LG-MC) for the 1D geometry shown in Fig. 1. For simplicity, the volume sources, S_{input} and Q_{input} , were given in the whole domain and the momentum source $S_{m,coll}$ and the recycling coefficient R were set to be 0. To save the calculation cost, the electron temperature was not calculated, being set as $T_e = T_i$. Table 1 shows the parameters in the benchmark test. Boundary conditions at $x = 0$ m are set as $V_0 = 0$ m/s, $\partial T_i / \partial x = 0$.

To get a final result of each scheme, one should check time-convergence and mesh-convergence. This means that the final result should be stationary and should not change by using finer mesh. In addition, for LG and LG-MC, particle-convergence has to be checked as well, which means the final result should not change by increasing the number of particles. The sequence of the checking procedure is 1) time-convergence, 2) particle-convergence, and then 3) mesh-convergence.

For FV, we will show only the final result later. Here we first show results of the mesh-convergence test of LG which was done after confirming the time- and particle-convergence. For all the schemes, equidistant mesh was used. This means the length of a cell is simply defined by L_x / N_x , where N_x is the number of mesh in the domain. As shown in Fig. 5, there is almost no shift between the result with $N_x = 100$ and those with $N_x = 200$, which means it is already mesh-converged with $N_x = 100$. (The results of the flow velocity are missing just because the difference between the two cases is too small to see.)

To discuss particle-convergence, we define $N_p \equiv N_{particle} / N_x$, where $N_{particle}$ is an input parameter for LG and LG-MC. In LG and LG-MC, total number of the particles N_{total} is controlled in a range $N_{particle} / 2 \sim 2N_{particle}$

in the following manner; If $N_{total} > 2N_{particle}$, a half of the particles are randomly killed and the rest have a twice larger weight, which is so-called Russian roulette method [16]. With this method, conservations are statistically satisfied. If $N_{total} < N_{particle} / 2$, each particle is divided into two particles with a half value of the original weight. In Fig. 5, all the results are with $N_p = 8000$, because the particle-convergence was confirmed with $N_p = 8000$ in each case.

From here, we show results from the convergence tests of LG-MC. The mesh-convergence of LG-MC was confirmed by Fig. 6. Before checking the mesh-convergence, the particle-convergence of the each mesh case was tested. The results of the particle-convergence test for the case with $N_x = 200$ are shown in Fig. 7. Compared to LG ($N_p = 8000$), more particles were needed to obtain the particle-convergence.

To check time-convergence of the calculation, the relative difference of the profile f was defined as

$$\frac{df}{f} \equiv \left| \frac{f^{new} - f^{old}}{f^{old}} \right|. \quad (35)$$

Besides, the residual error Res_f was calculated. For example, Res_n of the continuity equation, Eq. (1), was defined as

$$Res_n \equiv S_n - \frac{\partial}{\partial x} (nV). \quad (36)$$

The values in Eqs. (35) and (36) were calculated in all the cells and integrated over the cells and averaged. Figures 8 and 9 show the time development of those values

Table 1 Parameters for the 1D benchmark.

Parameter	Value
L_x	5 m
S_{input}	$2.5 \times 10^{22} \text{ m}^{-3} \text{ s}^{-1}$
Q_{input}	0.18 MW m^{-3}
$S_{m,coll}$	$0 \text{ kg m}^{-2} \text{ s}^{-2}$
γ_i	3.5
V_n	$1.55 \times 10^4 \text{ m/s}$ (thermal velocity for 2.5 eV)
$E_{e,ion}$	-25.0 eV (including excitation energy)
R	0

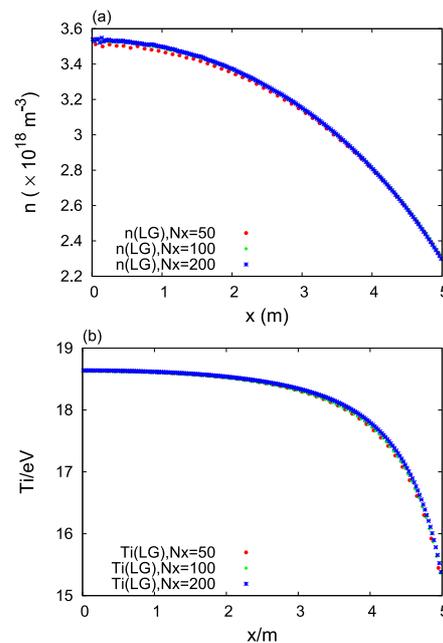


Fig. 5 Mesh-convergence of LG (Top: density, Bottom: ion temperature). $N_p = 8000$.

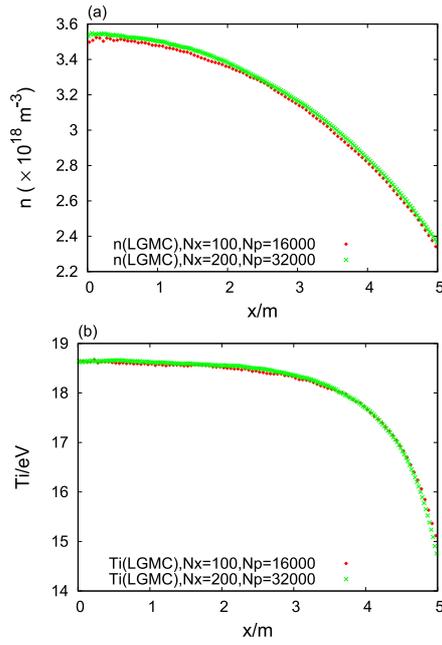


Fig. 6 Mesh-convergence of LG-MC (Top: density, Bottom: ion temperature).

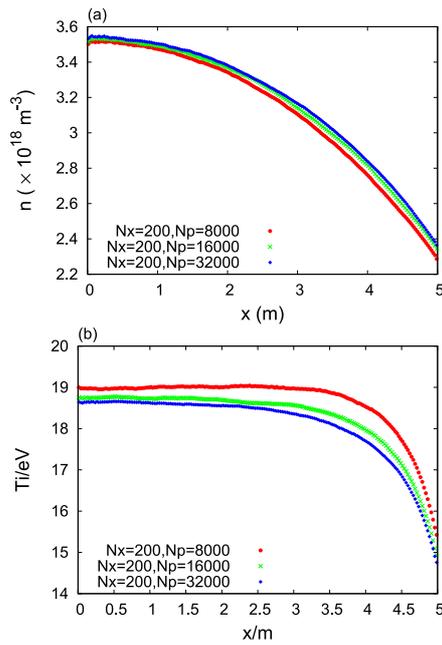


Fig. 7 Particle-convergence of LG-MC for the case $N_x = 200$ (Top: density, Bottom: ion temperature). N_p : number of particles in a cell.

for the case $N_x = 200, N_p = 32000$. From these results, we see that all the profiles are time-converged in the case. (Though we do not show here, the time-convergence was confirmed in this manner for all the cases.)

From Fig. 5 to Fig. 9, we obtained the final results of LG and LG-MC. To validate the final result of LG-MC ($N_x = 200, N_p = 32000$), we confirmed conservation laws

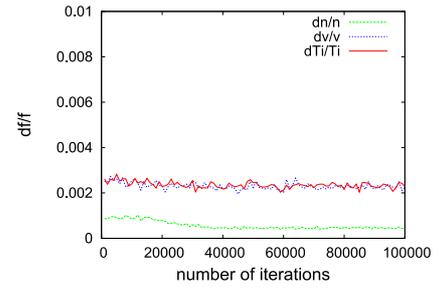


Fig. 8 Time development of the difference of the profile f in a time step df/f , defined in Eq. (35).

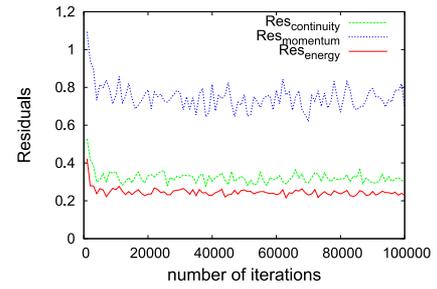


Fig. 9 Normalized Residuals of the profiles defined in Eq. (36).

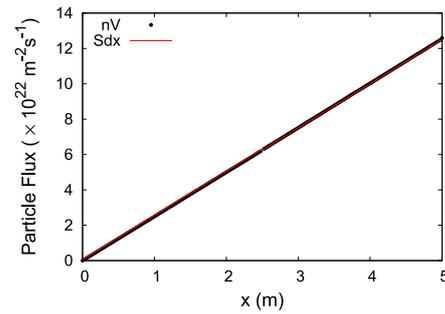


Fig. 10 Particle Conservation of the converged solution of LG-MC. Black points: particle flux, Red line: integration of the particle source term.

of the each equation, i.e., integrated form of Eqs. (1) - (3) (See Figs. 10 - 12).

Figure 13 shows the final results of the three schemes that are averaged in the last time steps. While FV and LG agreed well, a difference remained between LG and LG-MC.

To see the difference quantitatively, we define the difference of each profile f as follows.

$$\text{difference} \equiv \left| \frac{f_{\text{LG-MC}}(x) - f_{\text{LG}}(x)}{f_{\text{LG}}(x)} \right| \times 100[\%]. \quad (37)$$

As shown in Fig. 14, the difference is less than 5% in the entire calculation domain. The difference close to the boundary comes from the boundary setting of the conductive flux, which was explained in Sec. 3.6. Due to the mirror-reflection at the boundary, more energy particles are

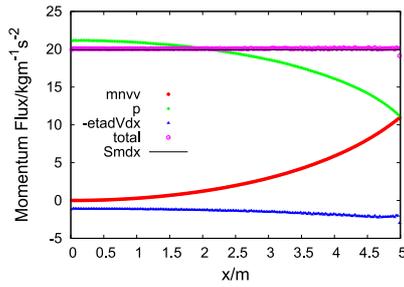


Fig. 11 Momentum Conservation of the converged solution of LG-MC. Red points: dynamic pressure, Green points: static pressure, Blue points: viscosity flux, Purple points: sum of the total pressure and the viscosity flux. Black line: integration of the momentum source.

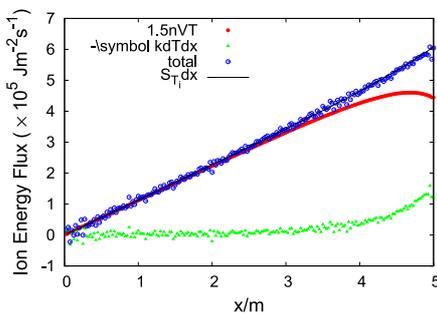


Fig. 12 Ion Energy Conservation of the converged solution of LG-MC. Red points: convective heat flux, Green points: conductive heat flux, Blue points: total heat flux, Black line: integration of the ion energy source term.

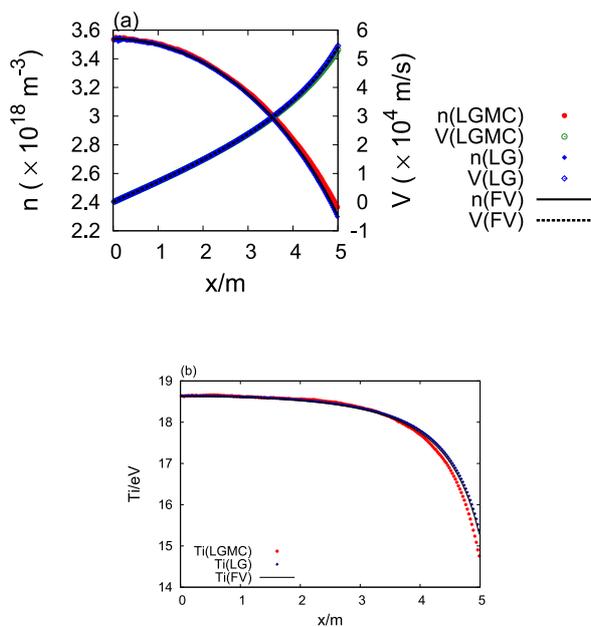


Fig. 13 Benchmark results of the three schemes, FV, LG, and LG-MC. Top: the density and the flow velocity, Bottom: the ion temperature. (LG: $N_x = 200$, $N_p = 8000$, LG-MC: $N_x = 200$, $N_p = 32000$)

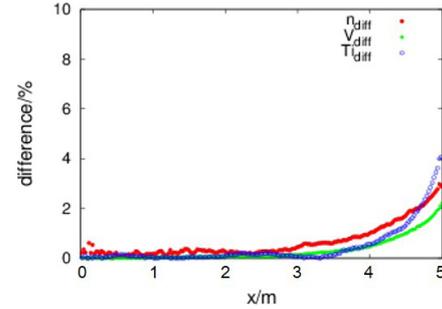


Fig. 14 Difference between the converged solutions of LG ($N_x = 200$, $N_p = 8000$) and LG-MC ($N_x = 200$, $N_p = 32000$).

accumulated near the boundary. It makes the energy outflux larger, making the temperature at the last cell lower.

Though this is a point to be improved, we confirmed basic validity of LG-MC.

5. Result of 3D LG-MC in a Cylinder

Adding the anomalous transports, the LG-MC code was extended to a simple 3D cylindrical geometry (See Fig. 15). The following equations were set, taking the z axis along the magnetic field. The anomalous transports were assumed to be by turbulence, thus they were put in the equations with the same coefficients both in the parallel and the perpendicular directions and their coefficients are independent each other.

$$\frac{\partial n}{\partial t} + \frac{\partial}{\partial z} \left(nV - D_a \frac{\partial n}{\partial z} \right) \quad (38)$$

$$+ \frac{\partial}{\partial x} \left(-D_a \frac{\partial n}{\partial x} \right) + \frac{\partial}{\partial y} \left(-D_a \frac{\partial n}{\partial y} \right) = S_n.$$

$$\frac{\partial}{\partial t} (mnV) + \frac{\partial}{\partial z} \left(mnV^2 + p - \frac{4}{3} \eta_i \frac{\partial V}{\partial z} - \eta_a \frac{\partial V}{\partial z} \right) \quad (39)$$

$$+ \frac{\partial}{\partial x} \left(-\eta_a \frac{\partial V}{\partial x} \right) + \frac{\partial}{\partial y} \left(-\eta_a \frac{\partial V}{\partial y} \right) = S_{m.coll.}$$

$$\frac{\partial}{\partial t} \left(\frac{3}{2} p_i \right) + \frac{\partial}{\partial z} \left(\frac{3}{2} p_i V - \kappa_i \frac{\partial T_i}{\partial z} - \kappa_{i,a} \frac{\partial T_i}{\partial z} \right) \quad (40)$$

$$+ \frac{\partial}{\partial x} \left(-\kappa_{i,a} \frac{\partial T_i}{\partial x} \right) + \frac{\partial}{\partial y} \left(-\kappa_{i,a} \frac{\partial T_i}{\partial y} \right) = S_{T_i}.$$

For the implementation, all the parallel transport terms (except the parallel particle diffusion) and the source terms were treated just as 1D LG without MC. From 1D LG, the following changes have been made for the 3D calculation;

1) The particles have 3D coordinates, (x, y, z) . After the convective transport in the z direction, each specie of the particles are transported by random walk steps in MC on the $x-y$ planes, according to the corresponding anomalous transport. The anomalous transport coefficients are given as $D_a = 1.0 \text{ m}^2/\text{s}$, $\eta_{i,a} = m_i n \times \chi_i$, and $\kappa_{i,a} = n \chi_i$, where $\chi_i = 0.2 \text{ m}^2/\text{s}$. All the quantities on the surface at $r = L_r$ were set as 0.

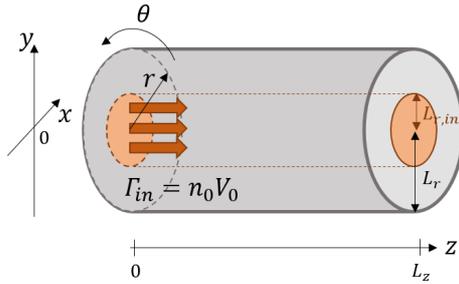


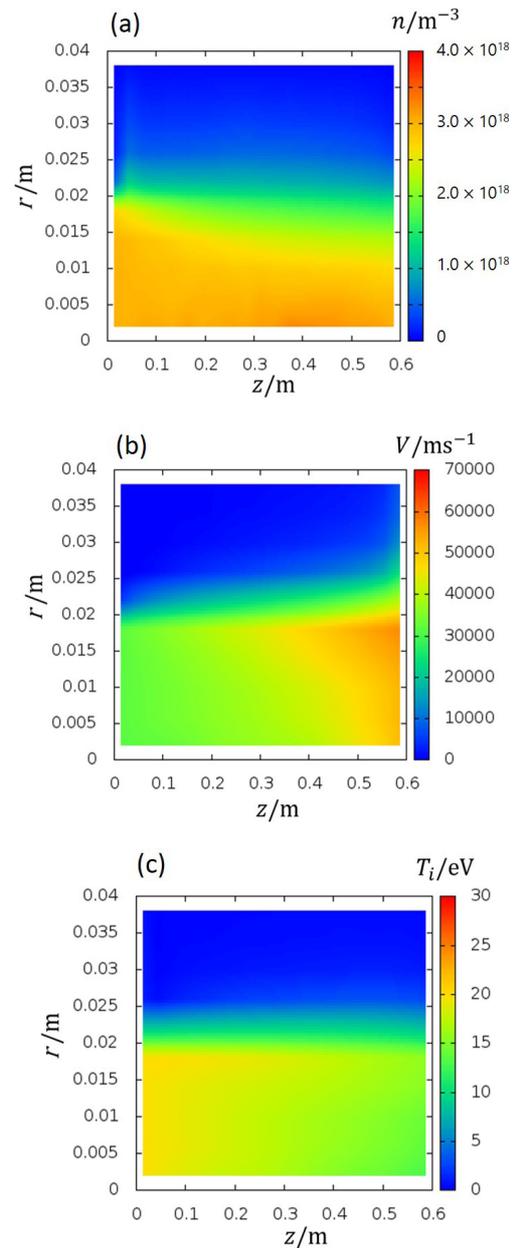
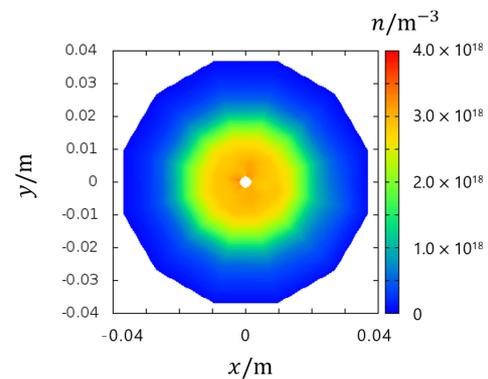
Fig. 15 3D cylinder model.

Table 2 Parameters for the 3D cylindrical case.

Parameter	Value
L_z	0.6 m
L_r	0.04 m
$L_{r,inf}$	0.02 m
R	0.5

2) Instead of the volume sources (S_{input} and Q_{input}), surface sources, i.e., fluxes of particle, momentum, and ion energy at $z = 0$ ($r = \sqrt{x^2 + y^2} \leq L_{r,inf}$) were given as $\Gamma(0) = n_0 V_0 = 3.0 \times 10^{18} \text{ m}^{-3} \times 0.5 \sqrt{2T_0/m}$, $\Pi(0) = m n_0 V_0^2$, and $q_i(0) = \frac{3}{2} n_0 V_0 T_0 = \frac{3}{2} \Gamma_0 \times 20 \text{ eV}$, respectively. To set these boundary conditions, the number of particles representing Γ_0 are added into the calculation domain from $z = 0 \text{ m}$ to $z = V_0 \Delta t$ with the weights $W_M = m W_n V_0$ and $W_{\epsilon_i} = \frac{3}{2} W_n T_0$.

Aiming at applying it to the D-module of GAMMA10/PDX [17] in the future, similar parameters from GAMMA10/PDX were chosen (See Table 2). Figure 16 shows results of the 3D LG-MC code in the cylindrical geometry. The flow velocity V increases near the wall ($z = 0.6 \text{ m}$) towards the ion sound speed (See Fig. 16(b)). If there was no particle source there, the increasing velocity makes the density lower, but due to the recycling source, the density keeps almost constant or slightly increases towards the wall (See Fig. 16(a)). It can also be seen that both density and velocity are diffused in the r direction. As for the ion temperature T_i , it decreases towards the wall ($z = 0.6 \text{ m}$) in the region where $r < 0.02 \text{ m}$. Figure 17 shows the density profile on the $x-y$ plane at $z = 0.3 \text{ m}$. It is confirmed from the color map that the plasma is isotropically diffused from the central region ($r < 0.02 \text{ m}$). Along the r coordinate (See Fig. 18), the density decays logarithmically in the outer region ($r > 0.02 \text{ m}$), which qualitatively agrees with the analytic solution of the diffusion equation in the r -coordinate. In the central region, due to the incoming parallel particle flux, the decay is slight. As for the velocity in the central region, it increases towards $r > 0$ because of decrease of the density. In the outer region, since $\eta_{i,a}/m;n < D_a$, the decay length for the velocity is shorter than the density.


 Fig. 16 Results of the 3D LG-MC code (r - z plane). (a) density, (b) flow velocity, (c) ion temperature.

 Fig. 17 Result of density in the $x-y$ plane at $z = 0.3 \text{ m}$.

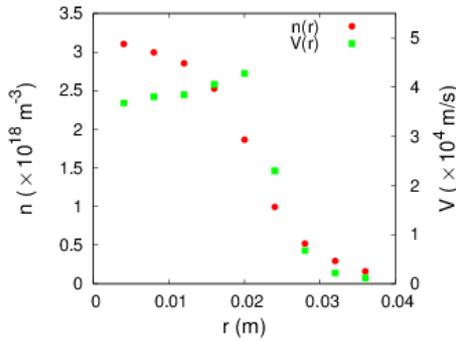


Fig. 18 Result of density along the r -axis at $z = 0.3$ m.

6. Conclusion

We developed a 3D SOL/Divertor plasma fluid code with the Lagrange-Monte-Carlo scheme (LG-MC). The use of test particles enables more flexible choice of numerical cells, compared to a conventional Finite Volume scheme (FV). This is the advantage of LG-MC for 3D complex geometries. Through the benchmark test in a simple 1D geometry with a FV code, validity of the LG-MC code has been confirmed. Comparing LG and LG-MC, LG seems to be better for the 1D test case along a magnetic field line, because it has less noise and shows faster convergence on the number of test particles. However, for future applications with ergodic magnetic fields for example, MC has a strong advantage because it can easily handle diffusion also in very complex meshes just by random walks. As the first test for a 3D case, LG-MC was implemented to a cylindrical geometry with anomalous transports. We obtained qualitatively reasonable results, which ensures fundamental usability of LG-MC in 3D geometries. Even though the test case was essentially a 2D calculation because of the axial symmetry, extension of the case to an asymmetric case can be easily done.

We plan to apply the code to the divertor simulation experimental module (D-module) of GAMMA10/PDX. For that aim, the next steps are planned as to 1) solve the electron temperature, 2) implement the real magnetic field configuration and the V-shaped target, 3) add source/sink terms from atomic/molecular processes (charge-exchange, recombination) into the transport equations, and 4) integrate with a kinetic neutral model.

Appendix. Weight of the Lagrange Particles

We define two coordinates along one axis. One is the Eulerian coordinate, x , and the other one is the Lagrange coordinate on a marker particle, ξ . As a function of ξ and the time t , the position on the Eulerian coordinate $x(\xi, t)$, the density $n(\xi, t)$, and the velocity $V(\xi, t)$ are set.

The motion of the marker is described as

$$\frac{dx(\xi, t)}{dt} = V(x, t). \quad (\text{A.1})$$

By differentiating the both sides with ξ , one obtains

$$\frac{\partial}{\partial \xi} \frac{dx(\xi, t)}{dt} = \frac{\partial V(x, t)}{\partial \xi} = \frac{\partial V(x, t)}{\partial x} \frac{\partial x}{\partial \xi}, \quad (\text{A.2})$$

which turns into

$$\left(\frac{\partial x}{\partial \xi}\right)^{-1} \frac{\partial}{\partial \xi} \frac{dx(\xi, t)}{dt} = \frac{\partial V(x, t)}{\partial x}. \quad (\text{A.3})$$

Since ξ is not a function of t , the left-side-hand (LHS) can be replaced to obtain

$$\left(\frac{\partial x}{\partial \xi}\right)^{-1} \frac{d}{dt} \frac{\partial x(\xi, t)}{\partial \xi} = \frac{\partial V(x, t)}{\partial x}. \quad (\text{A.4})$$

Substituting this into the form,

$$\frac{df}{dt} = -f \frac{\partial V}{\partial x}, \quad (\text{A.5})$$

we find

$$\frac{df}{dt} = -f \left(\frac{\partial x}{\partial \xi}\right)^{-1} \frac{d}{dt} \frac{\partial x(\xi, t)}{\partial \xi}, \quad (\text{A.6})$$

which leads

$$\frac{d}{dt} \left(f \frac{\partial x}{\partial \xi} \right) = 0. \quad (\text{A.7})$$

Here, the derivative $\frac{\partial x}{\partial \xi}$ has a simple geometrical meaning: it is the Jacobian describing squeezing or stretching of the initial fluid volume, therefore the weight of the particle has the form $f \frac{\partial x}{\partial \xi}$. Equation (A.7) shows that the weight is constant. Therefore, we can conclude that the weight for f should not be changed to take into account the term $-f \frac{\partial V}{\partial x}$ on RHS of Eq. (A.5).

Acknowledgments

The author R. T would like to thank Dr. R. Kanno (National Institute of Fusion Science), Mr. S. Kemnitz (University of Greifswald), and Mr. L. Lewerentz (University of Greifswald), for dedicated support and thoughtful discussions. This work was supported by Grant-in-Aid for JSPS (Japan Society for the Promotion of Science) Research Fellows, Grant Number 18J12692.

- [1] B.J. Braams, NET Report No. **68**, January 1987 (EUR-FU/XII-80/87/68).
- [2] R. Schneider *et al.*, Contrib. Plasma Phys. **46**, 3 (2006).
- [3] A.S. Kukushkin *et al.*, Fusion Eng. Des. **86**, 2865 (2011).
- [4] X. Bonnin *et al.*, Plasma Fusion Res. **11**, 1403102 (2016).
- [5] K. Shimizu *et al.*, Nucl. Fusion **49**, 065028 (2009).
- [6] T.D. Rognlien *et al.*, J. Nucl. Mater. **196-198**, 347 (1992).
- [7] M.S. Islam *et al.*, Plasma Phys. Control. Fusion **59**, 125010 (2017).
- [8] A.M. Runov *et al.*, Phys. Plasmas **8**, 916 (2001).
- [9] A. Runov *et al.*, Nucl. Fusion **44**, S74 (2004).
- [10] Y. Feng *et al.*, J. Nucl. Mater. **812**, 266 (1999).
- [11] G. Kawamura *et al.*, Contrib. Plasma Phys. **54**, 437 (2014).
- [12] R. Tatsumi *et al.*, Contrib. Plasma Phys. **56**, 516 (2016).

- [13] R. Tatsumi *et al.*, *Contrib. Plasma Phys.* **58**, 675 (2018).
- [14] S.I. Braginskii, "Transport processes in a plasma." *Reviews of plasma physics* **1**, 205 (1965).
- [15] C.W. Gardiner, *Handbook of Stochastic Methods for Physics, Chemistry and the Natural Sciences* (Springer, Berlin, 1985).
- [16] J. Spanier and E.M. Gelbard, *Monte Carlo Principles and Neutron Transport Problems* (Dover Publications, NY, 2008).
- [17] Y. Nakashima *et al.*, *Nucl. Fusion* **57**, 116033 (2017).