ITB Graduate School Conference (IGSC) ISSN: 2963-718X
December 21, 2021

Particle-in-Cell (PIC) Solver Using Least Squares Moving Particle Semi-implicit (LSMPS) Method

Edwin Aldrian Santoso[1], Luqman Fathurrohim[1] & Lavi Rizki Zuhal[1]

[1]Faculty of Mechanical and Aerospace Engineering, Bandung Institute of Technology
Email: 23620021@mahasiswa.itb.ac.id

Abstract. Particle-in-Cell (PIC) is the most common method used in a fully-kinetic solver, usually for low-density gases and non-equilibrium plasma simulation. The finite volume method is the natural choice for solving the Eulerian field because of its familiarity and ability to simulate arbitrary geometries. This work introduces the least squares moving particle semi-implicit (LSMPS) method as an alternative by using the particle method for solving spatial differentiation in Particle-in-Cell solver. In order to conduct a full simulation, some essential supporting functions should also be made, such as the field weighting for the Lagrangian particle description and the particle pusher algorithm to accommodate the external forces, such as Lorentz force in plasma. Plasma oscillation benchmarking will be used to validate all of the functions.

Keywords: *LSMPS; Particle-in-Cell; PIC; Plasma; Plasma oscillation.*

1 Introduction

Particle-in-Cell (PIC), as mentioned by Birdsall and Langdon in [1], is a method that combines the Lagrangian description of fluid particles with the Eulerian description of field variables on a mesh. This approach is suitable for fully-kinetic simulation, where many particles are required to be simulated. The computational cost could be lower by implementing the field solver on the mesh, and later the values are weighted to the particle and vice versa. Fully-kinetic simulation is used for low-density fluid numerical simulation because the continuum approach is not applicable due to the non-equilibrium nature of its microstate. In other words, the mean-free-path length is comparable or larger than the domain size, which implies that the Maxwellian distribution has not been achieved for the velocity distribution function (VDF). The VDF could be resolved by using many particles as the sample in the simulation.

Plasma is one of the applications that can be simulated numerically using the PIC solver due to its low particle density. Plasma is the combination of charged particles in the form of electrons, ions, and non-charged particles, neutrals. Some attempts to simulate plasma have been conducted using PIC solver, for example, by Szabo and Sanchez in [2] for fully kinetic simulation, and by Olshevsky et al. in [3] for fluid-PIC simulation. Also, some recent works to optimize PIC solver

have been conducted, for example, by Juhasz et al. in [4]. Another recent method of simulating plasma uses the direct kinetic method by solving the Boltzmann equation by Hara et al. in [5] and Raisanen et al. in [6]. This method has less numerical noise but also has a higher computational cost.

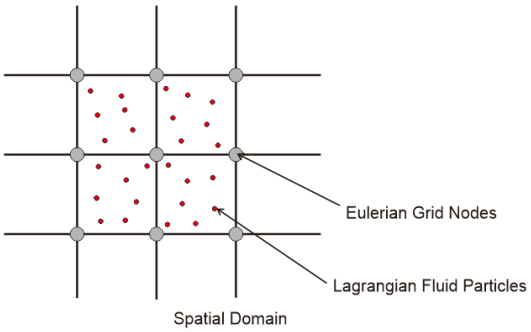

Figure 1 PIC Illustration.

The fluid particles used in PIC numerical simulation are superparticles, which means each particle represents a number of real atomic particles due to computation capability. As shown in Error! Reference source not found., these particles are called the Lagrangian fluid particles, and they move through the spatial domain during simulation. For each time step, the particles' properties are weighted to the Eulerian grid nodes. Later, the updated values from solving the field equation are weighted back to the particles and used to move the Lagrangian fluid particles. This PIC algorithm is visualized in Error! Reference source not found..

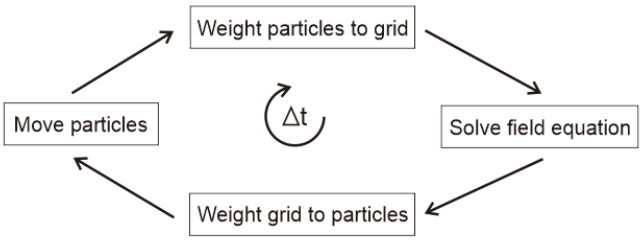

Figure 2 PIC Flowchart.

The LSMPS method, which was developed by Koshizuka and Tamai in [7], is a method to solve PDE equations with the Lagrangian approach. The popularity of LSMPS has underlined some implementations on fluid dynamics simulation like in [8] by Gao et al. This work proposed LSMPS to solve the field equation in Eulerian description replacing the FVM. However, the grid nodes used in this

work are still the same, although LSMPS can solve the field equation with randomized nodes position. The purpose here is to make sure that the LSMPS method can work well in PIC solver.

2 Theoretical Background

Many supporting functions are required to build a complete PIC solver. Based on the basic flowchart in **Error! Reference source not found.**, there are three main functions: weighting, solving field equations (Poisson's equation), and moving particles (Lorentz force). Without considering collisions terms, only special cases of plasma conditions can be simulated. In this section, a description of each function is given.

2.1 Weighting

There are two types of weighting: scattering, from particles to grid, and gathering, from the grid to particles. The weighting used is linear weighting based on the location of the particle inside the cell. The proportion in the horizontal direction is α_1, and in the vertical direction is α_2. The color area shows the scattering weighting proportions to the four nodes in **Error! Reference source not found.**. It is also the same principle for gathering. This method of weighting was proposed by Birdsall and Langdon in [1].

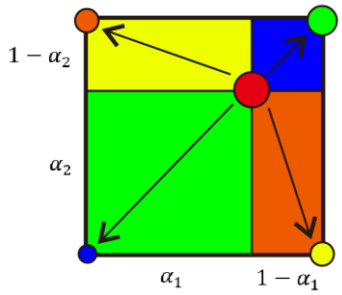

Figure 3 Weighting for Scattering.

If the Eulerian grid is not in rectilinear configuration, then bi-linear mapping should be implemented first to calculate the value of α_1 and α_2 or following the weighting method proposed by Seldner and Westermann in [9].

2.2 LSMPS

The least squares moving particle semi-implicit (LSMPS) method was first published by Koshizuka and Tamai in [7] to simulate incompressible flow with free surfaces. The spatial derivatives used are obtained by minimizing the error

using the weighted least squares function. This LSMPS procedure will be used in this work to solve the field equation in the PIC solver.

The expression of derivatives ($D_x f$) in each particle location (x_i) is determined by the value (f) of its neighbor particles (x_j) under distance less than the sphere of influence (r_e) as shown in **Error! Reference source not found.**, the set of the neighbor members is Λ_i. The formulation LSMPS is based on Taylor series expansion, so the polynomial basis (p) for the derivatives are in order, like in two dimensions becomes $[1, x, y, xy, x^2, y^2]$. In the formulation, the variables are also divided by a scaling parameter (r_s), which is chosen between 0 and r_e.

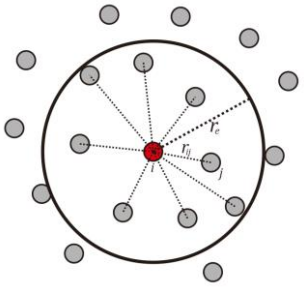

Figure 4 LSMPS Neighboring Particles.

The weight function (ω) can be chosen arbitrarily, as long as the weight function value is finite, its value outside the sphere of influence is zero, and its value decreases when it is farther from the origin point. In the simulation, this weight function is used.

$$\omega(x, r_e) = \begin{cases} \left(1 - \frac{|x|}{r_e}\right)^4 , 0 \le |x| < r_e \\ 0 \quad , \ |x| > r_e \end{cases} \tag{1}$$

The formulation for the derivatives Eq. (2) is shown in the following equations, where α is the order of the polynomial basis.

$$D_x f(x_i) = H_{r_s} M_i^{-1} b_i \tag{2}$$

$$H_{r_s} \equiv diag \left\{ \left\{ r_s^{-|\alpha|} \alpha! \right\}_{1 \le |\alpha| \le p} \right\} \tag{3}$$

$$M_i \equiv \sum_{j \in \Lambda_i} \left(\omega(x_j - x_i, r_e) p\left(\frac{x_j - x_i}{r_s}\right) p^T \left(\frac{x_j - x_i}{r_s}\right) \right) \tag{4}$$

$$b_i \equiv \sum_{j \in \Lambda_i} \left(\omega(x_j - x_i, r_e) p\left(\frac{x_j - x_i}{r_s}\right) \{ f(x_j) - f(x_i) \} \right) \tag{5}$$

2.3 Poisson's Equation

The equation that needs to be solved in the Eulerian grid nodes for plasma simulation is Poisson's Equation. The charge density value (ρ) is obtained from the particles' charge scattering to the nodes, and its value is updated every time step. The Poisson's equation for electrostatic ($\nabla \times E = 0$), no induced magnetic field, is given by Eq. (6) where ϕ is electric potential, and ε_0 is the electric permittivity in a vacuum. These formulations are taken from [10] by Griffiths.

$$\nabla^2 \phi = -\frac{\rho}{\varepsilon_0} \tag{6}$$

This equation is solved using LSMPS spatial derivation by applying appropriate boundary conditions in the domain. The electric field (E) is the value needed for moving the Lagrangian particles by Lorentz force. Therefore, the value of the electric field is obtained from the negative gradient of electric potential, also using LSMPS spatial derivation operator.

$$\mathrm{E} = -\nabla \phi \tag{7}$$

2.4 Lorentz Force

The acceleration of the Lagrangian particles is obtained from Lorentz force (F_L) due to electric charge (q) behavior under electric (E) and magnetic (B) fields.

$$\mathrm{F_L} = q(E + v \times B) \tag{8}$$

The relation of force and acceleration can be easily obtained from Newton's Second Law, but a relativistic approach is used here $\mathrm{F} = \frac{d(\gamma m v)}{dt}$, where $\gamma = \frac{1}{\sqrt{1-(v/c)^2}}$ is the Lorentz factor.

$$m\frac{d(\gamma v)}{dt} = q(E + v \times B) \tag{9}$$

The updated velocities are numerically obtained using the method by Boris in [11], but the method by Vay in [12] is used for the relativistic pusher algorithm due to the γ factor. The updated positions are obtained using the standard Leap-frog algorithm.

2.5 Plasma Oscillation

Plasma oscillation is the rapid oscillation of electron density due to charge imbalance, although the total charge in the domain is zero. This phenomenon will be simulated to validate the full PIC solver using the LSMPS method. The

oscillation can be observed from the change of total electric potential energy (E_p) in the Eulerian grid nodes and total electrons' kinetic energy (E_k).

$$E_p = \sum_{nodes} \left(\frac{1}{2} \varepsilon_0 E^2 \Delta V \right) \tag{10}$$

$$E_k = \sum_{electrons} \left((\gamma - 1) m_e c^2 \right) \tag{11}$$

The nodes' volume (ΔV) is the area around the nodes for a two-dimension domain, usually is chosen as the area of the quadrilateral of cell center points of the surrounding cells. The plasma angular frequency (ω_p) is given in Eq. (12), where n_e is electron number density and e is the electron's elementary charge. The formulation is taken from [13] by Chen.

$$\omega_p = \sqrt{\frac{n_e e^2}{m_e \varepsilon_0}} \tag{12}$$

The observed energy oscillation frequency should be twice the plasma frequency because electrons are at rest twice before returning to their original positions as a complete cycle.

3 Validation Results

In this work, three validations are conducted: the LSMPS method of solving Poisson's equation, the particle pusher algorithm, and the plasma oscillation validation. The validation domains of particle pusher and plasma oscillation are taken from the work of Kuhn and Groll in [14].

3.1 LSMPS solver and FVM Solver

In order to check the credibility of the LSMPS method of solving Poisson's equation, an analytical problem in Eq. (13) with an analytical solution in Eq. (14) is used as the benchmarking problem. Also, an FVM solver is made as a comparison to the result of the LSMPS method. The domain is square in xy plane [-1,1] and has the Dirichlet boundaries for all sides, which are equal to 0.

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = -2\pi^2 \sin(\pi x) \sin(\pi y) \tag{13}$$

$$f(x, y) = \sin(\pi x) \sin(\pi y) \tag{14}$$

In this simulation, r_s is set as the distance between nodes, and r_e is set as two times and a half of r_e. The results are compared to the analytical result in terms

of Root Mean Square Error, RMSE = $\sqrt{\frac{\sum_{i=1}^{N}(f_p - f_a)^2}{N}}$, where f_p is the predicted value, f_a is the analytical value, and N is the number of nodes.

Table 1 RMSE of LSMPS and FVM.

Domain	LSMPS	FVM
25 x 25	7.485e-03	2.539e-03
50 x 50	1.912e-03	6.456e-04
100 x 100	4.840e-04	1.629e-04

Both LSMPS and FVM can accurately solve Poisson's equation, as seen from the small RMSE values. Although LSMPS accuracy is lower than FVM, this method is still good as an alternative to FVM.

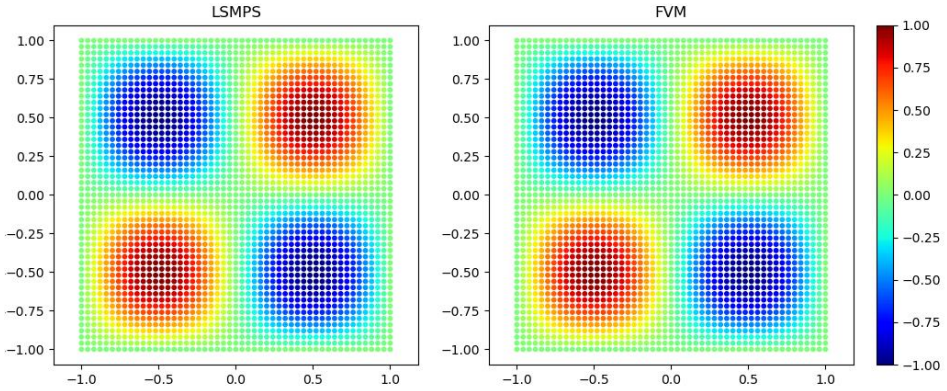

Figure 5 LSMPS and FVM Results (50 x 50).

3.2 Particle Pusher

The algorithm of moving the Lagrangian fluid particles contains two parts: parallel acceleration due to the electric field and perpendicular acceleration due to the magnetic field. Each of the acceleration will be validated separately.

3.2.1 Electric Acceleration

The setup is a one-dimensional domain, where there is an electric potential difference (U) between the ends. An electron with mass m_e and charge e is placed at zero velocity at one end, then the electron is accelerated by the electric field, and the final velocity (v_f) at the end is captured. From the conservation of energy, the analytical result of the final velocity is given by Eq. (13).

$$v_f = c\sqrt{1 - \frac{1}{\left(1+\frac{eU}{m_e c^2}\right)^2}} \tag{15}$$

Both the Boris method and Vay's method are validated here compared to the analytical result, with a variation of electric potential differences from $1\,V$ to $10^6\,V$. As shown in Error! Reference source not found., the results in the non-relativistic regime are accurate for both solvers. Also, as expected, when the electric potential difference is high and the final velocity is approaching the speed of light (c), the relativistic particle pusher using Vay's method can still get an accurate result.

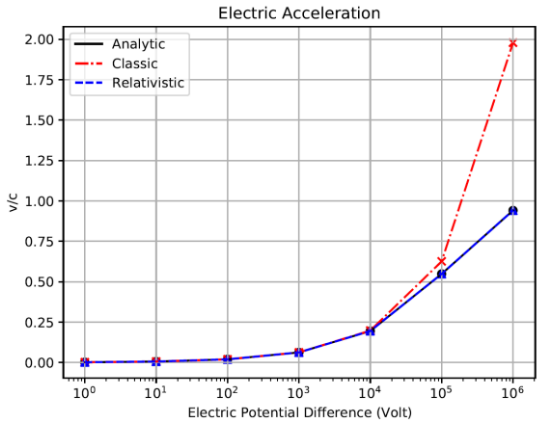

Figure 6 Electric Acceleration Validation.

3.2.2 Magnetic Circular Motion

Constant magnetic field (B = 1 T) is applied in the z-direction, and an electron is placed on xy plane with initial velocity ($v_0 = 0.1c$) in the x-direction at a position gyro radius (r_g) below the origin. The electron should be moving in a circular motion around the origin and has angular frequency same as gyro angular frequency (ω_g).

$$r_g = \frac{ev_0}{\gamma m_e} \tag{16}$$

$$\omega_g = \frac{eB}{\gamma m_e} \tag{17}$$

The electron's trajectory is a circular motion with $r_g = 17.12\,mm$ as shown in **Error! Reference source not found.**. The theoretical frequency of the circular

motion is 27.88 GHz, and in the simulation is 27.85 GHz. Overall, the particle pusher algorithm is doing well.

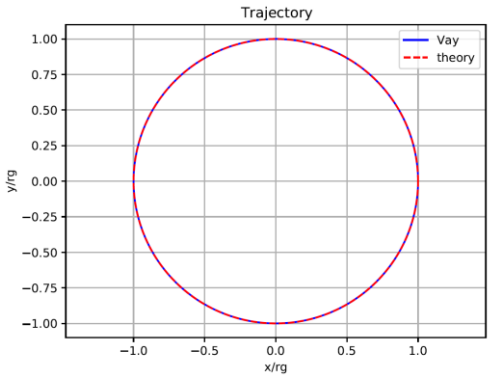

Figure 7 Magnetic Circular Motion Validation.

3.3 Plasma Oscillation

This benchmark validates the combination of the particle weighting algorithm in the PIC method, the particle pusher algorithm using Vay's method, and Poisson's equation solver using the LSMPS method. The domain size is $6\,m \times 0.8\,m$, discretized into 120×16 cells, and all boundaries are grounded into 0 V. For this test, one million zero-velocity electrons are scattered inside the domain with appropriate particle weight, so the electron number density is $n_e = 10^{13}\ m^3$, and the same number of ions are placed but at slightly different positions by $\Delta x = 0.01 \sin(2\pi x/3)$ from the electrons' positions. In addition, ions are fixed at their positions and not allowed to move.

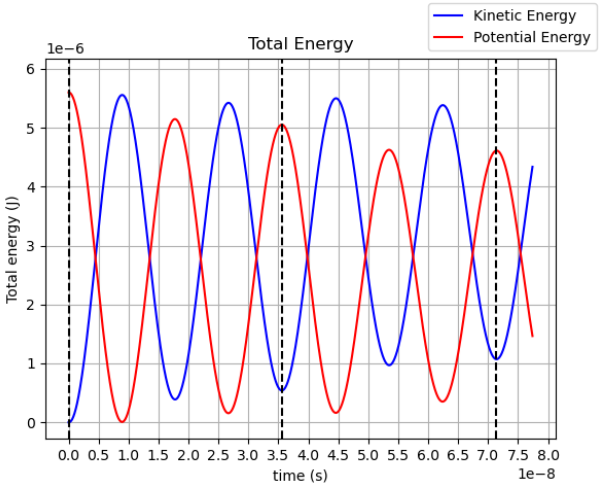

Figure 8 Plasma Oscillation Validation.

The electrons will move by the electric field induced by the displacement at the beginning. After that, the energy will be converted into the electron's kinetic energy. The electrons will stop again, and the energy is transferred back to electric potential energy. As a result, oscillation behavior in the total energy is observed, as shown in Error! Reference source not found.. The first complete oscillation is achieved after $35.58\,ns$ since the simulation started, and the second is after $71.38\,ns$. In short, the average period of the oscillation in the simulation is $35.69\,ns$, and the theoretical value of the period based on Eq. (12) is $35.20\,ns$.

Overall, the plasma oscillation can be captured well, and the frequency value is just slightly different from the theoretical value. Moreover, the energy is conserved during the simulation oscillating back and forth between kinetic and electric potential energy. However, the electrons in the simulation could not be back to their initial rest conditions, and they still have some kinetic energy remaining.

4 Conclusions

In conclusion, the LSMPS method is doing well and compatible with the PIC solver. Although the LSMPS method's accuracy is lower than the FVM in the same Eulerian grid configuration for solving Poisson's equation, this method has the potency to be a great method in the future. One example is a fully particle-based simulation where the Eulerian grid nodes do not need to be in a structural configuration.

The particle pusher algorithm can resolve the motion of the particles accurately under Lorentz force. The parallel accelerated motion due to the electric field is very close to the theoretical prediction. It is also the case for perpendicular accelerated motion due to the magnetic field. Relativistic particle pusher using Vay's method is proven to be accurate and can be used for PIC simulation in the future.

This work has validated the plasma oscillation, and the plasma frequency obtained is close to the theoretical value. However, some numerical errors appear in the result. It leads to the inability of the electrons to be back to their original positions, as can be seen from the total kinetic energy that keeps increasing even though the total energy in the simulation is conserved. These numerical errors come from the accumulated weighting error and the LSMPS numerical error.

Different weighting schemes for scattering and gathering between the Eulerian grid nodes and the Lagrangian fluid particles can be analyzed in future work. The LSMPS method has parameters that can be changed, such as the weighting

function and the radius of influence (r_e). These parameters can be optimized to every simulated domain to achieve minimum numerical error.

For further works and applications, the function to handle collisions should be added to the solver, as mentioned by Bird in [15]. By having collisions models as the addition, an actual domain such as electric propulsion devices, which was listed by Goebel and Katz in [16], can be numerically simulated using the developed solver.

Acknowledgments

We thank the Faculty of Mechanical and Aerospace Engineering, Bandung Institute of technology, for providing the High-Performance Computer (HPC) facility, which helps us conduct this work more efficiently and perform the numerical simulation faster.

References

[1] Birdsall, C. & Langdon, A., *Plasma Physics via Computer Simulation*, Taylor & Francis, 2005.
[2] Szabo, J.J. & Sanchez M.M., *Fully Kinetic Hall Thruster Modeling*, International Electric Propulsion Conference (IEPC), **341**, 2001.
[3] Olshevsky, V.., *Slurm: Fluid Particle-in-Cell Code for Plasma Modeling*, Computer Physics Communications, **235**, 2019.
[4] Juhasz, Z. et al., *Efficient GPU Implementation of the Particle-in-Cell/Monte-Carlo Collisions Method for 1D Simulation of Low-pressure Capacitively Coupled Plasmas*, Computer Physics Communications, 263, 2021.
[5] Hara K. et al. *One-dimensional Hybrid-direct Kinetic Simulation of the Discharge Plasma in a Hall Thruster*, Physics of Plasmas, **19**, 2012.
[6] Raisanen, A.L. et al. *Two-dimensional Hybrid-direct Kinetic Simulation of a Hall Thruster Discharge Plasma*, Physics of Plasmas, **26**, 2019
[7] Koshizuka, S. & Tamai, T., *Least squares moving particle semi-implicit method*, Computational Particle Mechanics (CPM), **1**, pp. 277-305, 2014.
[8] Gao W. et al., *A Coupled 3D Isogeometric/Least-square MPS Approach for Modeling Fluid–structure Interactions,* Computer Methods in Applied Mechanics and Engineering, **373,** 2021.
[9] Seldner, D. & Westermann, T., *Algorithms for Interpolation and Localization in Irregular 2D Meshes*, Journal of Computational Physics, **79**, pp. 1-11, 1988.
[10] Griffiths, D.J., *Introduction to Electrodynamics*, ed. 3, Prentice Hall, 1998.
[11] Boris, J.P., *The Acceleration Calculation from A Scalar Potential*, Plasma Physics Laboratory, Princeton University, March 1970.

[12] Vay, J.L., *Simulation of Beams or Plasmas Crossing at Relativistic Velocity*, Physics of Plasmas, **15**, 2008.
[13] Chen, F.F., *Introduction to Plasma Physics and Controlled Fusion*, ed. 3, Springer, 2016.
[14] Kuhn, C. & Groll, R., *picFoam: An OpenFOAM Based Electrostatic Particle-in-Cell Solver*, Computer Physics Communications, **262**, 2021.
[15] Bird, G., *Molecular Gas Dynamic and The Direct Simulation of Gas Flows*, Oxford University Press, 1994.
[16] Goebel, D. M. & Katz, I., *Fundamentals of Electric Propulsion: Ion and Hall Thrusters*, John Wiley and Sons, 2008.