# Coupling multi-fluid dynamics equipped with Landau closures to the particle-in-cell method

**Rouven Lemmerz** [1,2,†], **Mohamad Shalaby** [1,†], **Timon Thomas** [1] **and Christoph Pfrommer** [1]

[1]Leibniz-Institut für Astrophysik Potsdam (AIP), An der Sternwarte 16, 14482 Potsdam, Germany

[2]University of Potsdam, Institute of Physics and Astronomy, Karl-Liebknecht-Str. 24-25, 14476 Potsdam, Germany

The particle-in-cell (PIC) method is successfully used to study magnetized plasmas. However, this requires large computational costs and limits simulations to short physical run times and often to set-ups of less than three spatial dimensions. Traditionally, this is circumvented either via hybrid-PIC methods (adopting massless electrons) or via magneto-hydrodynamic-PIC methods (modelling the background plasma as a single charge-neutral magneto-hydrodynamical fluid). Because both methods preclude modelling important plasma-kinetic effects, we introduce a new fluid-PIC code that couples a fully explicit and charge-conserving multi-fluid solver to the PIC code SHARP through a current-coupling scheme and solve the full set of Maxwell's equations. This avoids simplifications typically adopted for Ohm's law and enables us to fully resolve the electron temporal and spatial scales while retaining the versatility of initializing any number of ion, electron or neutral species with arbitrary velocity distributions. The fluid solver includes closures emulating Landau damping so that we can account for this important kinetic process in our fluid species. Our fluid-PIC code is second-order accurate in space and time. The code is successfully validated against several test problems, including the stability and accuracy of shocks and the dispersion relation and damping rates of waves in unmagnetized and magnetized plasmas. It also matches growth rates and saturation levels of the gyro-scale and intermediate-scale instabilities driven by drifting charged particles in magnetized thermal background plasmas in comparison with linear theory and PIC simulations. This new fluid-SHARP code is specially designed for studying high-energy cosmic rays interacting with thermal plasmas over macroscopic time scales.

## 1. Introduction

Astrophysical plasmas naturally partition into thermal and non-thermal particle populations. Provided particles collide frequently via (Coulomb) collisions, this eventually

† Email addresses for correspondence: rlemmerz@aip.de, mshalaby@live.ca

leads to a characteristic thermal Maxwellian phase-space distribution. This population can be reliably described with the fluid approximation, which characterizes a vast number of particles by a few macroscopic fields in space (e.g. number density, mean velocity and temperature). By contrast, the non-thermal cosmic ray (CR) ion population at energies exceeding GeV is mostly collisionless and interacts with the background plasma via wave–particle interactions, thus retaining its initial power-law distribution for much longer times (Blandford & Eichler 1987; Draine 2011; Zweibel 2017). Low-energy CRs ($\lesssim$ GeV) more frequently experience Coulomb/ionization collisions and as such have a direct influence on the gas dynamics and molecular chemistry (Dalgarno 2006; Padovani *et al.* 2020). The CRs can excite and grow plasma waves via instabilities at which they scatter in pitch angle (i.e. the angle between momentum and magnetic field vector), thereby regulating their macroscopic transport speed and exchanging energy and momentum with the thermal population. Modelling these plasma processes requires us to move beyond the classical fluid approximation.

During the process of diffusive shock acceleration, CRs stream ahead of the shock into the precursor region and drive non-resonant Alfvén waves unstable by means of their powerful current (Bell 2004; Riquelme & Spitkovsky 2009; Caprioli & Spitkovsky 2014*a*), which provides efficient means of increasing their wave–particle scattering and reducing the CR diffusion coefficient (Caprioli & Spitkovsky 2014*b*). Upon escaping from the acceleration site into the ambient medium, CRs continue to drive Alfvén waves through resonant instabilities. Scattering off of these self-induced waves regulates their transport speed (Kulsrud & Pearce 1969; Marcowith, van Marle & Plotnikov 2021; Shalaby, Thomas & Pfrommer 2021), which is determined by the balancing instability growth and wave damping (Thomas & Pfrommer 2019; Thomas, Pfrommer & Enßlin 2020). In the interstellar medium, CRs provide a comparable if not dominant pressure, despite their negligible number densities in comparison with the thermal population, which makes them dynamically important (Boulares & Cox 1990; Draine 2011). Their pressure gradient can drive outflows from the interstellar medium (Simpson *et al.* 2016; Farber *et al.* 2018; Girichidis *et al.* 2018) so that powerful global winds emerge from galaxies (Uhlig *et al.* 2012; Hanasz *et al.* 2013; Pakmor *et al.* 2016; Ruszkowski, Yang & Zweibel 2017) that enrich the circumgalactic medium in galaxy haloes with CRs that can also dominate the pressure support and modify the cosmic accretion of gas onto galaxies (Buck *et al.* 2020; Ji *et al.* 2020). The degree to which CRs regulate galaxy formation critically depends on the efficiency of wave–particle interactions, which in turn depend on the amplitude of self-excited plasma waves (Thomas, Pfrommer & Pakmor 2022). On even larger scales, CRs energized in jets of active galactic nuclei stream into the surrounding intracluster medium of cool core clusters and heat it via the excitation of Alfvén waves and the successive damping (Guo & Oh 2008; Pfrommer 2013; Jacob & Pfrommer 2017; Ruszkowski, Yang & Reynolds 2017). Because the plasma physics underlying these processes is highly nonlinear, numerical calculations are needed to study these effects.

Due to its ability to resolve kinetic processes, the particle-in-cell (PIC) method (Dawson 1962; Langdon & Birdsall 1970; Hockney 1988; Birdsall & Langdon 1991) has become one of the most used methods for studying plasmas from laboratory to astrophysical scales. Examples of that include revolutionizing our understanding of the rich physics found in collisionless shocks (Spitkovsky 2008; Marcowith *et al.* 2016), magnetic reconnection (Daughton, Scudder & Karimabadi 2006; Daughton *et al.* 2011; Sironi & Spitkovsky 2014), instabilities driven by highly relativistic electron–positron beams (Bret, Gremillet & Dieckmann 2010; Shalaby *et al.* 2017*a*, 2018, 2020) as well as the transport of non-thermal particle populations like CRs (Holcomb & Spitkovsky 2019; Shalaby *et al.* 2021).

However, the PIC method needs to advance numerous particles per cell each time step, and thus it is quick to reach its computational limit. Even one-dimensional simulations usually only capture the dynamics on very short physical times and the extent to which two- or three-dimensional simulations can be performed is very limited.

The time interval between the inverse of the electron plasma frequency, $\omega_e^{-1}$, (which is necessary to ensure the stability of the PIC algorithm) and that of the ion plasma frequency, $\omega_i^{-1}$, depends on the ion-to-electron mass ratio, since $\omega_i^{-1}/\omega_e^{-1} = (m_i/m_e)^{1/2}$, assuming charge neutrality, i.e. that the electron and ion densities are equal. Therefore, one frequently used trick to increase the computational efficiency in PIC simulations is to adopt a reduced ion-to-electron mass ratio to bridge the gap between the smallest time scale in the simulation and the larger time scale on which interesting physical processes occur. However, this might lead to artificial suppression of physical effects (Bret & Dieckmann 2010; Hong *et al.* 2012; Moreno *et al.* 2018), including instabilities with excitation conditions that depend on the mass ratio (Shalaby *et al.* 2021, 2022). This shows the need for a more efficient numerical method to complement the accurate results achieved by PIC simulations in order to enable simulations of realistic physics occurring on longer time scales. One possible method consists in using the less expensive fluid approximation, which works particularly well for collisional systems where frequent particle collisions maintain a thermodynamic temperature but is less well motivated in weakly collisional or even collisionless astrophysical plasmas where it cannot accurately capture some important microphysical plasma processes.

Multiple methods have been devised that combine the computational advantages of a fluid code, while trying to maintain some of the physics accuracy provided by the PIC method. Hybrid-PIC codes (Lipatov 2002; Gargaté *et al.* 2007) treat electrons as a massless fluid and ions as particles. With the assumption of charge neutrality and the Darwin approximation (i.e. neglecting the transverse displacement current), these codes are able to overcome some computational barriers while omitting effects on the electron time and length scale. Since this eliminates the need to resolve electron scales, the increase in computational efficiency from pure-PIC to hybrid-PIC methods is roughly a factor of $(m_i/m_e)^{1/2}$ in time scale and approximately the same factor in spatial scales. In cases where the electron pressure anisotropy becomes important, such as in magnetic reconnection, a hybrid Vlasov–Maxwell system can be coupled to an anisotropic electron fluid with a Landau fluid closure, which captures more kinetic physics (Finelli *et al.* 2021). On the other hand, an even more efficient method exists, that combines the magneto-hydrodynamic (MHD) description of the thermal background plasma with PIC methods to model the evolution of energetic particles such as CRs (Bai *et al.* 2015; van Marle, Casse & Marcowith 2018), called MHD-PIC. However, this method inherits the assumptions of MHD, in particular, the use of (a simplified) Ohm's law by fully neglecting the displacement current, which precludes physics associated with the higher-order terms of Ohm's law as well as the electron dynamics.

In this paper we present a self-consistent algorithm that is suitable for simulating microphysical effects of CR physics by only applying the fluid approximation to thermal particles and solving the full set of Maxwell's equations. Our goal of this novel fluid-PIC method is to sacrifice as little physics accuracy as possible, while at the same time alleviating computational restraints by orders of magnitude for set-ups involving CRs (or similar, low density non-thermal particle populations interacting with a thermal plasma). The fluid-PIC method, in essence, couples a multi-fluid solver to the PIC method by summing their contributions to the charge and current densities used to solve Maxwell's equations, and the resulting electromagnetic fields. Thus, the subsequent dynamics is dictated by fluid and PIC species. This enables the treating of any arbitrary number

of species in thermal equilibrium by modelling them as separate fluids that interact electromagnetically with each other and with particles of arbitrary momentum distribution (modelled using the PIC method). In contrast to MHD-PIC and hybrid-PIC methods, we do not explicitly assume Ohm's law, and instead solve Maxwell's equations in a fully self-consistent manner in our fluid-PIC code. Therefore, displacement currents are included in our model and fast changes in the electric field and electron dynamics are captured. This, in turn, allows us to study the interaction of high-energy particles with the background plasma, e.g. to investigate CR streaming. Another hybrid approach resolving electron time scales fully, but using pressure coupling, has been used for the simulation of pick-up ions in the heliosphere by Burrows, Ao & Zank (2014).

Often implicit and semi-implicit methods are utilized for stability and resolution reasons to couple the multi-fluid equations to Maxwell's equations (Hakim, Loverich & Shumlak 2006; Shumlak *et al.* 2011; Wang *et al.* 2020). However, this creates an interdependency between all fluids and has limited utility when coupled to explicit particles. We have developed an explicit multi-fluid solver in which each fluid and particle species is agnostic about each other and the coupling is achieved via an indirect current-coupling scheme. Because the PIC part of the code is the most computationally expensive part of the fluid-PIC, hybrid-PIC and MHD-PIC methods, the computational efficiency is mostly determined by the number of particles required as well as the smallest time and length scales that need to be resolved. Hence, this fluid-PIC approach results in large speed ups for CR propagation simulations in comparison with traditional hybrid-PIC codes, which treat every ion as a particle and need to initialize a large number of particles according to the density ratio, as well as in comparison with PIC-only simulations. Especially studying CR propagation in the interstellar medium, where the typical CR density is of the order of $10^{-9}$ times the interstellar medium number density, is challenging. Since the fluid-PIC algorithm is faster by orders of magnitude in comparison with PIC in such a case, we can reach further into the realistic parameter regime without sacrificing some essential microphysics.

One of the most important kinetic effects is arguably Landau damping. The fluid description can emulate this effect using Landau closures (Hammett & Perkins 1990; Umansky *et al.* 2015; Hunana *et al.* 2019*a*), which necessitates the computation of the heat flux in Fourier space. While Fourier transforms in one dimension are not easily parallelizable, this bottleneck can partially be mitigated by performing global communications of the message-passing interface (MPI) in the background while processing the high computational load (e.g. resulting from evolving orbits of PIC particles) in the foreground. Simulations with periodic boundary conditions are currently handled by convolution with a finite-impulse-response (FIR) filter in our code, but other options are available in the literature (Dimits, Joseph & Umansky 2014; Wang *et al.* 2019). A number of simplifying local approximations exist as well (Wang *et al.* 2015; Allmann-Rahn, Trost & Grauer 2018; Ng *et al.* 2020), which scale computationally well but become inaccurate for studying some multiscale plasma physics problems. Our code implements these different approaches so that an appropriate one can be chosen, dependent on the requirements of a simulation. Our implementation is massively parallelized and can be efficiently run on thousands of cores. Furthermore, the fluid-PIC method allows for any multi-fluid set-up. As such, this framework allows for some straightforward extensions. Potentially, this involves a set-up with actively participating neutrals to incorporate ion–neutral damping into this method. To this end, the coupling between different fluids needs to be extended by a collision term, which is left as a future extension to the code.

The outline of this paper is as follows. In § 2, we introduce the pillars of this method and describe the PIC method, the fluid solver, how we couple both methods by means of electromagnetic fields and describe various implementations of the Landau closure. In § 3, we show validation tests of the fluid solver (shock-tube tests), linear waves in an ion–electron plasma and the damping rate of Langmuir waves in a single-electron fluid with Landau closures. We then investigate the nonlinear effects of two interacting Alfvén waves as well as CR-driven instabilities, where fluid-PIC and PIC results are compared. We conclude in § 4. Throughout this work, we use the SI system of units.

## 2. Numerical Method

After a review of the kinetic description of a plasma in § 2.1, we briefly introduce our PIC method in § 2.2. The fluid description for plasmas and its assumptions are given in § 2.3. The finite volume scheme we use to numerically solve the compressible Euler equations is described in § 2.4, while the electromagnetic interactions of the fluid are described in § 2.5. In § 2.6, we describe the Landau closure we adopt in order to mimic the Landau damping in kinetic thermal plasmas within the fluid description, and detail its implementation in our code. We close this section by describing the overall code structure of the fluid-PIC algorithm and finally discuss the interaction between the modules via the current-coupling scheme (§ 2.7).

### 2.1. *Kinetic description of a plasma*

The kinetic description of a collisionless relativistic plasma with particles of species $s$ with elementary mass, $m_s$, and elementary charge, $q_s$, is given by the Vlasov equation

$$\frac{\partial f_s}{\partial t} + \frac{\boldsymbol{u}}{\gamma} \cdot \nabla f_s + \boldsymbol{a}_s \cdot \nabla_u f_s = 0, \tag{2.1}$$

where $f_s = f_s(\boldsymbol{x}, \boldsymbol{v}, t)$ is the distribution function, $\boldsymbol{u} = \gamma \boldsymbol{v}$ is the spatial component of the four-velocity with the Lorentz factor $\gamma = [1 + (\boldsymbol{v}/c)^2]^{-1/2}$ and $c$ is the light speed. The acceleration due to the Lorentz force is given by

$$\boldsymbol{a}_s = \frac{q_s}{m_s} [\boldsymbol{E}(\boldsymbol{x}, t) + \boldsymbol{v} \times \boldsymbol{B}(\boldsymbol{x}, t)], \tag{2.2}$$

where $\boldsymbol{E}(\boldsymbol{x}, t)$ and $\boldsymbol{B}(\boldsymbol{x}, t)$ are the electric and magnetic fields, respectively. The evolution of electric and magnetic fields is governed by Maxwell's equations

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\nabla \times \boldsymbol{E}, \quad \nabla \cdot \boldsymbol{B} = 0, \tag{2.3}$$

$$\frac{\partial \boldsymbol{E}}{\partial t} = c^2 \nabla \times \boldsymbol{B} - \frac{\boldsymbol{J}}{\varepsilon_0}, \quad \nabla \cdot \boldsymbol{E} = \frac{\rho}{\varepsilon_0}, \tag{2.4}$$

where $c = 1/\sqrt{\varepsilon_0 \mu_0}$ is the vacuum speed of light, and $\varepsilon_0$ and $\mu_0$ are the permittivity and the permeability of free space, respectively. The evolution of the electromagnetic fields is influenced by the charge density, $\rho$, and current density, $\boldsymbol{J}$. They are given by the charge-weighted sum over all species of the number densities $n_s$ and bulk velocities $\boldsymbol{w}_s$,

respectively,

$$\rho(\boldsymbol{x}, t) = \sum_s q_s n_s(\boldsymbol{x}, t) = \sum_s q_s \int f_s(\boldsymbol{x}, \boldsymbol{v}, t) \, \mathrm{d}^3 v, \tag{2.5}$$

$$\boldsymbol{J}(\boldsymbol{x}, t) = \sum_s q_s n_s(\boldsymbol{x}, t) \boldsymbol{w}_s(\boldsymbol{x}, t) = \sum_s q_s \int \boldsymbol{v} f_s(\boldsymbol{x}, \boldsymbol{v}, t) \, \mathrm{d}^3 v. \tag{2.6}$$

### 2.2. *The PIC method*

We use the PIC method to solve for the evolution of plasma species that are modelled with the kinetic description. The PIC method initializes a number of computational macroparticles to approximate the distribution function in a Lagrangian fashion. Each macroparticle represents multiple physical particles and, as such, each macroparticle has a shape in position space which can be represented by a spline function. By depositing the particle motions and positions onto the numerical grid (or computational cells), the electromagnetic fields can be computed. This step is followed by a back interpolation of these fields to the particle positions so that the Lorentz forces on the particles can be computed. In our implementation, these equations are solved using one spatial dimension and three velocity dimensions (1D3V), i.e. $\nabla = (\partial/\partial x, 0, 0)^{\mathrm{T}}$.

The code quantities are defined as multiples of the fiducial units given for time, fields (electric and magnetic), charge, current density and length

$$\left.\begin{array}{c} t_0 = \sqrt{m_0 \epsilon_0/(q_0^2 n_0)}, \quad E_0 = \sqrt{n_0 m_0 c^2/\epsilon_0}, \\ \rho_0 = q_0 n_0, \quad J_0 = \rho_0 c, \quad x_0 = c t_0. \end{array}\right\} \tag{2.7}$$

This enables us to select a fixed time step of

$$\Delta t = C_{\mathrm{cfl}} c \Delta x, \tag{2.8}$$

where $C_{\mathrm{cfl}} < 0.5$ to satisfy the Courant–Friedrichs–Lewy (CFL) condition. The value of the reference density $n_0$ is chosen such that the code time scale, $t_0$, obeys $\omega_p^{-2} = t_0^2$. The total plasma frequency is $\omega_p = (\sum_s \omega_s^2)^{1/2}$, and is related to the plasma frequencies of the individual species, $\omega_s^2 = q_s^2 n_s/(m_s \epsilon_0)$. We define the discretized time $t^k = k \Delta t$, position $x_i = i \Delta x$ and quantities at discrete position and times as $E_i^k = E(t^k, x_i)$. For details on the PIC code SHARP, the reader is referred to Shalaby *et al.* (2017b, 2021). Here, we focus on describing how SHARP is extended to include fluid treatment of some plasma species.

### 2.3. *Fluid description of plasma*

A straightforward way of coarse graining the Vlasov equation (2.1) is to reduce its dimensionality. By taking the $j$th moment over velocity space, i.e. $\int \boldsymbol{v}^j f \, \mathrm{d}^3 v$, we retrieve the fluid quantities and reduce the dimensionality of the 1D3V kinetic description to one dimensions. The number density $n_s$ and the bulk velocity $\boldsymbol{w}_s$ are defined through the zeroth and first moments of the distribution function, respectively, while the total energy density per unit mass $\epsilon_s$ and the scalar pressure per unit mass $p_s$ are related to the second moment (Wang *et al.* 2015)

$$n_s(\boldsymbol{x}, t) = \int f_s(\boldsymbol{x}, \boldsymbol{v}, t) \, \mathrm{d}^3 v, \tag{2.9}$$

$$\boldsymbol{w}_s(\boldsymbol{x}, t) = \int \frac{1}{n_s(\boldsymbol{x}, t)} \boldsymbol{v} f_s(\boldsymbol{x}, \boldsymbol{v}, t) \, \mathrm{d}^3 v, \tag{2.10}$$

$$\epsilon_s(\pmb{x}, t) = \int \frac{1}{2} v^2 f_s(\pmb{x}, \pmb{v}, t) \, \mathrm{d}^3 v, \tag{2.11}$$

$$p_s(\pmb{x}, t) = \int (v_x - w_{s,x})^2 f_s(\pmb{x}, \pmb{v}, t) \, \mathrm{d}^3 v = \frac{\Gamma - 1}{2} \int (\pmb{v} - \pmb{w}_s)^2 f_s(\pmb{x}, \pmb{v}, t) \, \mathrm{d}^3 v. \tag{2.12}$$

Here, the pressure tensor is under the adiabatic assumption and the degrees of freedom are encoded in the adiabatic index $\Gamma$. The following relation is found from the definitions:

$$\epsilon_s = \frac{p_s}{\Gamma - 1} + \frac{1}{2} n_s \pmb{w}_s \cdot \pmb{w}_s. \tag{2.13}$$

The first three moments of the Vlasov equation are called the continuity, momentum and energy conservation equations. A set of these equations is found for each fluid species, but the subscript $s$ is neglected here for simplicity

$$\frac{\partial n}{\partial t} + \nabla \cdot (n\pmb{w}) = 0, \tag{2.14}$$

$$\frac{\partial n\pmb{w}}{\partial t} + \nabla \cdot [p\pmb{1} + n\pmb{w}\pmb{w}] = \frac{q}{m} \pmb{S}_w(n, \pmb{w}, \pmb{B}, \pmb{E}), \tag{2.15}$$

$$\frac{\partial \epsilon}{\partial t} + \nabla \cdot [(p + \epsilon)\pmb{w}] + \frac{1}{\Gamma - 1} \nabla \cdot \pmb{Q} = \frac{q}{m} \pmb{w} \cdot \pmb{S}_w(n, \pmb{w}, \pmb{B}, \pmb{E}). \tag{2.16}$$

We assumed the non-relativistic limit and an isotropic pressure tensor with vanishing non-diagonal components, i.e. the inviscid limit. The notation $\pmb{w}\pmb{w}$ indicates the dyadic product of the two vectors and $\pmb{1}$ is the unit matrix. Similar to the definition of the scalar pressure in (2.12) we use a definition of the heat flux vector, which is normalized to the degrees of freedom as well

$$\pmb{Q}(\pmb{x}, t) = \frac{\Gamma - 1}{2} \int (\pmb{v} - \pmb{w})^2 (\pmb{v} - \pmb{w}) f(\pmb{x}, \pmb{v}, t) \, \mathrm{d}^3 v. \tag{2.17}$$

The electromagnetic source term is given by

$$\pmb{S}_w(n, \pmb{w}, \pmb{B}, \pmb{E}) = n(\pmb{E} + \pmb{w} \times \pmb{B}). \tag{2.18}$$

The general form of the fluid equations can be written as

$$\frac{\partial \tilde{U}}{\partial t} + \nabla \cdot \pmb{F}(\tilde{U}) = \pmb{S}(\tilde{U}), \tag{2.19}$$

where $\tilde{U} = \tilde{U}(\pmb{x}, t) = (n, n\pmb{w}, \epsilon)^{\mathrm{T}}$ is the fluid state vector at position $(\pmb{x}, t)$, $\pmb{F}$ is the flux matrix and $\pmb{S}$ is the source vector.

Numerically, the complexity of solving (2.19) can be reduced by splitting the operator into less complex sub-operators using Strang operator splitting (Strang 1968; Hakim *et al.* 2006). This enables us to use the most appropriate solver for each subsystem sequentially. We split the fluid update into three parts: the flux $\pmb{F}$ excluding the heat flux (see § 2.4), the electromagnetic source $\pmb{S}_{\mathrm{em}} = \pmb{S}_w q/m$ (see § 2.5.1) and the heat flux $\pmb{Q}$ (see § 2.6). For commuting operators $\exp(\Delta t \pmb{Q})$ and $\exp(\Delta t \pmb{S}_{\mathrm{em}})$ a second-order accurate Strang splitting is obtained as

$$U^{n+1/2} = \mathrm{e}^{(\Delta t/2)\pmb{F}} \mathrm{e}^{\Delta t \pmb{Q}} \mathrm{e}^{\Delta t \pmb{S}_{\mathrm{em}}} \mathrm{e}^{(\Delta t/2)\pmb{F}} U^{n-1/2} + O(\Delta t^3). \tag{2.20}$$

If $\pmb{Q}$ and $\pmb{S}_{\mathrm{em}}$ act independently on the entries $p$ and $\pmb{w}$, respectively, then the order of applying them can be varied and they need to be evaluated only once. In practice, the

formulation of $Q$ might partially depend on $w$. In this case, Strang splitting is performed on this part of the operator $Q$ as well, see (2.47). In order to apply $S_{em}$ (and $Q$, which depends on the direction of $B$) one needs to find the electromagnetic quantities at time $t^n$ first. The components of $E$ along the simulated spatial direction can only be updated from time $t^{n-1}$ to $t^n$ after applying $\exp(\Delta t/2 \times F)$ for the first time (see § 2.5.2). Therefore, electromagnetic quantities need to be calculated between these updates. This is unproblematic as $F$ is independent of the electromagnetic field, and we can defer updating $E$ without reducing accuracy.

## 2.4. *Finite volume scheme*

The 1D3V fluid equations are solved using a finite volume method, where the fluid equations are averaged over the cell volume, which is an interval of length $\Delta x$ in one dimension,

$$U_i(t) = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} \tilde{U}(x, t)\, dx. \tag{2.21}$$

This enables us to correctly conserve the overall fluid mass, fluid momentum and fluid energy, even in the presence of large gradients, by utilizing Gauss' theorem

$$\frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} \frac{\partial F(\tilde{U})}{\partial x}\, dx = \frac{1}{\Delta x}[F_{i+1/2} - F_{i-1/2}], \tag{2.22}$$

where the flux through an interface at $x_i$ is $F_i(t) = F[\tilde{U}(x_i, t)]$, leading to the update equation

$$\frac{\partial U_i(t)}{\partial t} = \frac{1}{\Delta x}\left[-F_{i+1/2} + F_{i-1/2} + \int S(\tilde{U}(x, t))\, dx\right]. \tag{2.23}$$

Integrating equation (2.23) in time is achieved by using at least second-order Runge–Kutta methods (Butcher 2016), which is the limit set by the operator splitting scheme. We could not find examples yet, where higher-order Runge–Kutta methods have performed noticeably different from second-order methods in the fluid-PIC code. In contrast to the finite difference scheme used for electromagnetic fields and particles, where electromagnetic quantities are point values, fluid quantities discretized with the finite volume method are cell averages. This is useful, because the finite difference method does not guarantee the conservation of the conservation equations (2.14) through (2.16), which are governing the fluid; while on the other hand using the finite volume method for the electromagnetic fields needs additional steps to satisfy the constraint $\nabla \cdot B = 0$. Hybridization of both schemes to combine the advantages of each has been used before in other contexts, i.e. Soares Frazao & Zech (2002).

The maximum time step in the 1D3V Euler equations, which allows for stable simulations, is $\Delta t < C_{cfl}\Delta x/(|w| + c_s)$, with the speed of sound $c_s = (\Gamma p/n)^{1/2}$. For all realistic set-ups these velocities are limited naturally by the speed of light, $|w| < c$ and $c_s < c$, and this condition is automatically fulfilled by the time step criterion in (2.8). In practice, only (2.8) together with a suitable Courant number of $C_{cfl} \leq 0.5$ is used to determine the time step of the simulation.

### 2.4.1. *Reconstruction*

To approximate the flux at interfaces, we need to reconstruct the fluid state at cell interfaces. The accuracy of the reconstruction has a crucial influence on the diffusivity.

A lower-order reconstruction can lead to excessive damping of waves, which might suppress relevant physical effects on longer time scales.

For reconstructing the point value $\tilde{U}(x_{i+1/2}, t)$, which is needed to compute $F_{i+1/2}$, we employ a central-weighted essentially non-oscillatory reconstruction (C-WENO) scheme of spatial order five. The reconstruction computes two point values at each interface $x_{i+1/2}$, an interpolation from the left- and right-hand sides. We reconstruct the primitive variables $n$, $w$ and $p$ individually.

Our implementation of the C-WENO method is based on the fifth-order scheme presented in Capdeville (2008). An introduction to the topic can be found in Cravero *et al.* (2018*a*). The C-WENO reconstruction uses a convex combination of multiple low-order reconstruction polynomials to achieve high-order interpolations of the interface values while it employs a nonlinear limiter to degrade this high-order interpolation to a lower order if the reconstructed quantity contains discontinuities. The fifth-order C-WENO uses three third-order polynomials $P_L(x)$, $P_C(x)$, $P_R(x)$ for each cell $i$ to interpolate the four adjacent cells in the following way:

$P_L(x)$    interpolates values at    $i-2$    $i-1$     $i$,
$P_C(x)$    interpolates values at    $i-1$     $i$    $i+1$,
$P_R(x)$    interpolates values at      $i$    $i+1$    $i+2$,

while the optimal fifth-order polynomial interpolates all of them:

$P_{\mathrm{opt}}(x)$    interpolates values at    $i-2$    $i-1$    $i$    $i+1$    $i+2$.

We define an additional polynomial

$$P_0(x) = \frac{1}{d_0}\left[ P_{\mathrm{opt}}(x) - \sum_{q\in[L,C,R]} d_q P_q(x)\right], \tag{2.24}$$

where $d_0 + d_L + d_C + d_R = 1$. The polynomials $P_0$, $P_L$, $P_C$ and $P_R$ are a convex representation of the $P_{\mathrm{opt}}$ polynomial. We use $d_0 = 3/4$, $d_C = 2/16$ and $d_L = d_R = 1/16$.

In general, we would like to use the reconstruction provided by the $P_{\mathrm{opt}}$ polynomial as frequently as possible because of its high-order nature. But this high-order reconstruction can cause oscillations similar to the Gibbs phenomenon at discontinuities. Therefore, we need to employ a limiting strategy to avoid such behaviour. In order to accomplish this, we re-weight all of our $d$-coefficients by taking the smoothness of the associated polynomial into account (Jiang & Shu 1996). We define

$$\alpha_q = d_q\left[1 + \left(\frac{\tau}{\mathrm{IS}[P_q] + 10^{-9}\Delta x}\right)^2\right] \quad \text{for } q \in [0, L, C, R], \tag{2.25}$$

where $\tau$ is a measure for the overall smoothness of the reconstructed variables, and $\mathrm{IS}[P_q]$ defines a smoothness indicator of the low-order polynomials. Because the formulae for these smoothness indicators are quite cumbersome, we list them in Appendix A. These coefficients define a new set of normalized weights given by

$$w_q = \frac{\alpha_q}{\alpha_0 + \alpha_L + \alpha_C + \alpha_R} \quad \text{for } q \in [0, L, C, R]. \tag{2.26}$$

The final reconstructed polynomial is then given by the convex combination of the low-order polynomials using this set of normalized weights

$$P_{\mathrm{rec}}(x) = w_0 P_0(x) + w_L P_L(x) + w_C P_C(x) + w_R P_R(x), \tag{2.27}$$

which we evaluate at the cell interfaces to calculate the required left- and right-handed interface values for the Riemann solver. We detail how these polynomials are evaluated in Appendix A.

The smoothness indicators $\mathrm{IS}[P_q]$ vanish if the underlying polynomials are smooth. In this case, the re-weighted coefficients reduce to their original value $\alpha_q \to d_q$ and the reconstructed polynomial reduces to the optimal polynomial $P_{\mathrm{rec}}(x) \to P_{\mathrm{opt}}(x)$.

### 2.4.2. *Riemann solver*

The previous reconstruction step determines two, potentially different, values $\tilde{U}_L$ and $\tilde{U}_R$ for each quantity to the left and right of every interface, thereby providing the initial conditions for the Riemann problem

$$\frac{\partial U}{\partial t} = -\nabla \cdot \boldsymbol{F}(\tilde{U}), \tag{2.28}$$

$$\tilde{U}(x, 0) = \begin{cases} \tilde{U}_L & x < 0 \\ \tilde{U}_R & x > 0 \end{cases}. \tag{2.29}$$

An (approximate) Riemann solver is employed to compute the numerical flux $\boldsymbol{F}(\tilde{U})$. While a number of different families of Riemann solvers have been developed with individual strengths and weaknesses, we have decided to implement multiple solvers which can be changed on demand. Implemented solvers in fluid-SHARP include a Roe solver with entropy fix (Roe 1981; Harten & Hyman 1983) and an HLLC (Harten–Lax–van Leer contact) solver (Toro, Spruce & Speares 1994). While the Roe solver yields more accurate solutions and fewer overshoots in our tests in comparison with the HLLC solver, it becomes unstable in near vacuum flows and strong expansion shock waves. Even though differences between the solvers are easily visible in some shock set-ups and artificially extreme conditions, they are typically negligible in most applications common for thermal plasmas. We opt to employ the HLLC solver as our standard for stability purposes and use the Roe solver in cases where stronger shocks with overshoots are expected.

### 2.4.3. *Importance of wave characteristics in approximating stable numerical fluxes*

The characteristic curves of the Euler equations without sources correspond to the eigensystem of the flux Jacobian $\partial F/\partial U$. Approximate Riemann solvers use these characteristics for computing fluxes across small time steps and to introduce numerical dissipation to suppress spurious instabilities. For the hydrodynamic Euler equations in 1D3V (without a heat flux), five characteristics emerge with characteristic wave speeds $\lambda = w_x - c_s, w_x + c_s, w_x$, where the last eigenvalue $w_x$ has a multiplicity of 3. In particular, the Roe solver (without the entropy fix) computes the numerical flux at an interface by averaging the physical fluxes as follows (LeVeque 2002):

$$\boldsymbol{F}_{\mathrm{num}} = \tfrac{1}{2}[\boldsymbol{F}(\tilde{U}_L) + \boldsymbol{F}(\tilde{U}_R) - \boldsymbol{D}_\lambda(\tilde{U}_R, \tilde{U}_L)]. \tag{2.30}$$

The dissipation vector $\boldsymbol{D}_\lambda = \boldsymbol{R}|\boldsymbol{\Lambda}|\boldsymbol{R}^{-1}(\tilde{U}_R - \tilde{U}_L)$ vanishes for $\tilde{U}_R = \tilde{U}_L$. The matrices $\boldsymbol{R}(\tilde{U}_L, \tilde{U}_R)$ and $|\boldsymbol{\Lambda}|(\tilde{U}_L, \tilde{U}_R)$ are composed of eigenvectors and a diagonal of the absolute eigenvalues $|\lambda(\tilde{U}_L, \tilde{U}_R)|$, respectively, where an appropriate averaging of left- and right-hand states at the interface is used to derive these eigenvectors and eigenvalues. That is, the dissipation is directly based on the jump at the interface of each wave multiplied by its characteristic wave velocity. The dissipation vector satisfies the subcharacteristic condition, i.e. in characteristic coordinates each eigenvalue is bounded

by the dissipation $-D_{\lambda,i} \leq \lambda_i \leq D_{\lambda,i}$, and thus stabilizes the scheme without introducing excessive dissipation (Whitham 1974; Chen & Liu 1993; Hsiao 1997; LeVeque & Pelanti 2001). In multi-dimensional scenarios, computing the projection of the difference $(\tilde{U}_R - \tilde{U}_L)$ from a Cartesian grid onto the characteristics can result in violations of this condition or excessive dissipation. In these instances, it can be advantageous to artificially alter the wave speeds entering $D_\lambda$. Reduced wave speeds can be used to successfully counter excessive dissipation leading to a wrong convergence for low Mach number flows (Dellacherie 2010), however, this leads to numerical instabilities when applied to high Mach number flows. On the other hand, increased wave speeds have been found to eliminate numerical instabilities at shocks (Peery & Imlay 1988). While the HLLC solver makes more sophisticated approximations to the wave speeds of the nonlinear system, the principle of artificially increasing selected wave speed estimates yields the same result (Simon & Mandal 2019).

To provide an understanding of how these characteristics influence the operator splitting, suppose the following decomposition of the total flux $F_t = F_A + F_B$ into two fluxes. Hence, we need to compare the numerical estimate of the total flux with that of the individual subsystems, denoted by $F_A$ and $F_B$. The expansion of nonlinear systems provided by Strang (1968) yields $U_i^{n+1} = U_i^n + \Delta t[\Delta F_{i,A}(U^n) + \Delta F_{i,B}(U^n)] + O(\Delta t^2)$, where we only use terms up to first order for simplicity. The intracell flux $\Delta F_i = -(F_{\mathrm{num},i+1/2} - F_{\mathrm{num},i-1/2})/\Delta x$ is used for updating $U$ (see (2.23)). The numerical flux at an interface for one time step is thus

$$F_{\mathrm{num}} = \begin{cases} \frac{1}{2}[F_t(\tilde{U}_L) + F_t(\tilde{U}_R) - D_{\lambda,t}(\tilde{U}_R, \tilde{U}_L)] & \text{if unsplit} \\ \frac{1}{2}[F_t(\tilde{U}_L) + F_t(\tilde{U}_R) - D_{\lambda,A}(\tilde{U}_R, \tilde{U}_L) - D_{\lambda,B}(\tilde{U}_R, \tilde{U}_L)] & \text{if split} \end{cases}. \quad (2.31)$$

Both numerical fluxes converge to the total physical flux; for vanishing dissipation vectors, e.g. $\tilde{U}_R = \tilde{U}_L$, both formulations are equal. Intuitively, the total strength of the dissipation matrix in the unsplit scheme is smaller $\mathrm{tr}|\Lambda_t| \leq \mathrm{tr}|\Lambda_A| + \mathrm{tr}|\Lambda_B|$, while the split scheme is stable provided that the subsolvers are stable (Strang 1968). As an important consequence, a split Riemann solver only needs to account for the characteristics in the subsystem. This conveniently allows us to use specific solvers for each subsystem without taking into consideration the other systems of equations. Another possibility is to convert the divergence of a flux into a source term, which eliminates the need for a Riemann solver but results in the loss of guaranteed conservation in the finite volume scheme. We provide two applications, for which this is useful.

First, the heat flux vector $Q$, which is indeed a physical flux, results in a non-trivial change of the wave characteristics. Instead of including this complexity in the Riemann solver here, it is simpler to treat its divergence ($\nabla \cdot Q$) as a source term instead (see § 2.6).

Second, in the MHD limit, the evolution equation of the electromagnetic momentum $\epsilon_0 \partial_t E \times B = -\rho E - J \times B + \nabla \cdot T_{\mathrm{em}}$ collapses to the constraint $J \times B \sim \nabla \cdot (BB - 1B^2/2)/\mu_0$ (see, e.g. Braginskii 1965), where the Maxwell stress tensor is given by $T_{\mathrm{em}} = \epsilon_0 EE + BB/\mu_0 - 0.5(\epsilon_0 E^2 + B^2/\mu_0)1 \approx (BB - 1B^2/2)/\mu_0$. That is, the electromagnetic source term can be expressed as a divergence of an electromagnetic flux tensor. The MHD solvers make use of this constraint and, by including electromagnetic fluxes into their total flux, the MHD Riemann solvers add dissipation based on the full MHD wave characteristics. Failing to do so leads to numerical instabilities, especially for large magnetic field strengths, as in general the fast magnetosonic wave is faster than the characteristic waves treated in our scheme. However, because this source-flux equivalency is invalid without the MHD assumptions, we must include the Lorentz force

as a source term (see the following § 2.5.1) and consequently do not use the MHD characteristic velocities in our Riemann solver. In § 3.2, we demonstrate the accuracy of our implemented algorithm for propagating MHD waves, demonstrating that the numerical dissipation is sufficient to suppress potential numerical instabilities.

## 2.5. *Electromagnetic interaction with charged fluids*

In this section, we first introduce the Lorentz force as a source term in (2.15). Furthermore, we describe how the fluid influences the electromagnetic fields. With these two additional parts, the description of an uncharged gas in § 2.4 is expanded here to include plasmas.

### 2.5.1. *Treatment of electromagnetic source term*

Instead of integrating the energy equation (2.16), which would require evaluating the source term on the right-hand side, we convert $\epsilon$ into $p$ before applying the source update $\exp(\Delta t S_{\mathrm{em}})$ (see (2.20)). Consequently, we compute the time evolution of the primitive pressure variable, for which the electromagnetic source term conveniently vanishes

$$\frac{\partial p}{\partial t} + \Gamma p \nabla \cdot \boldsymbol{w} + \boldsymbol{w} \cdot \nabla p + \nabla \cdot \boldsymbol{Q} = 0. \tag{2.32}$$

Then, only the computation of the source term for the momentum equation (2.15) is left. Up until now we have only applied the C-WENO method for conservation laws, however, by adding the source term, we are left with a balance law. In C-WENO formulations for balance laws it is customary to approximate the integral of the source term ((2.23)) numerically to higher orders as well (Cravero *et al.* 2018*b*). We use Simpson's formula for approximating (2.23)

$$\int_{x_{i-1/2}}^{x_{i+1/2}} \boldsymbol{S}(\tilde{\boldsymbol{U}}) \, \mathrm{d}x = \tfrac{1}{6}(\boldsymbol{S}(\tilde{\boldsymbol{U}}_{i-1/2}) + 4\boldsymbol{S}(\tilde{\boldsymbol{U}}_i) + \boldsymbol{S}(\tilde{\boldsymbol{U}}_{i+1/2})) + O(\Delta x^5), \tag{2.33}$$

where the intracell values $\tilde{\boldsymbol{U}}_{i\pm1/2}$ are interpolated by the same C-WENO scheme as used for solving the hydrodynamical equations, and the centre value is computed self-consistently with the numerical integration formula, i.e. $\tilde{\boldsymbol{U}}_i = (6\boldsymbol{U}_i - \tilde{\boldsymbol{U}}_{i+1/2} - \tilde{\boldsymbol{U}}_{i-1/2})/4$. We also need to interpolate the electromagnetic field values to a comparable spatial order. This is achieved by performing finite difference interpolations for each component from the Yee mesh discretized fields, that is

$$\boldsymbol{E}_{i+1/2} = \frac{150(\boldsymbol{E}_i + \boldsymbol{E}_{i+1}) - 25(\boldsymbol{E}_{i-1} + \boldsymbol{E}_{i+2}) + 3(\boldsymbol{E}_{i-2} + \boldsymbol{E}_{i+3})}{256} + O(\Delta x^6), \tag{2.34}$$

and temporal order, $\boldsymbol{B}^n = (\boldsymbol{B}^{n+1/2} + \boldsymbol{B}^{n-1/2})/2$, again, for each component necessary. Lower-order approximations produce, in our tests, similar results, but converge to slightly lower wave frequencies when compared with the analytical solution of the dispersion relation. We apply $\boldsymbol{S}(\tilde{\boldsymbol{U}}_i)$ by using an implicit velocity update

$$\frac{\boldsymbol{w}_i^{n+1/2} - \boldsymbol{w}_i^{n-1/2}}{\Delta t} = \frac{q}{m}\left[\boldsymbol{E}_i^n + \frac{1}{2}(\boldsymbol{w}_i^{n+1/2} + \boldsymbol{w}_i^{n-1/2}) \times \boldsymbol{B}_i^n\right], \tag{2.35}$$

which is solved using the Boris velocity integration (Boris *et al.* 1970). The splitting of fluid flux update and Lorentz force ((2.20)) is reminiscent of pushing a particle with the PIC-method, where the Lorentz force for a full-time step is calculated in between half-time step updates of the position vector.

### 2.5.2. *Deposition of charges*

Equations (2.4) govern the electric field evolution, where Faraday's or Gauss' law might be used to compute $\boldsymbol{E}$. In this section we focus on the one-dimensional set-up without particle contributions, which are explained in §2.7. The perpendicular components' update, $E_y$ and $E_z$, is received straightforwardly by discretizing Faraday's law

$$(E_y)_{i+1/2}^{n+1} = (E_y)_{i+1/2}^n - \sum_s \frac{\Delta t}{\epsilon_0} q_s (nw_y)_{i+1/2,s}^{n+1/2} - \frac{c^2 \Delta t}{\Delta x}[(B_z)_{i+1}^{n+1/2} - (B_z)_i^{n+1/2}], \quad (2.36)$$

$$(E_z)_{i+1/2}^{n+1} = (E_z)_{i+1/2}^n - \sum_s \frac{\Delta t}{\epsilon_0} q_s (nw_z)_{i+1/2,s}^{n+1/2} + \frac{c^2 \Delta t}{\Delta x}[(B_y)_{i+1}^{n+1/2} - (B_y)_i^{n+1/2}], \quad (2.37)$$

where the sum is taken over all fluid species $s$ and $nw$ are components of the fluid vector $\boldsymbol{U}$.

For the $E_x$ component in spatial direction however, in order to enforce charge conservation, Gauss' law in discretized form needs to be enforced for all $i \geq 1$ as well

$$(E_x)_i^n = (E_x)_0^n + \sum_s \frac{q_s}{\epsilon_0} \sum_{j=0}^{i-1} n_{j+1/2,s}^n \Delta x = (E_x)_0^n + \sum_s \frac{q_s}{\epsilon_0} \int_{x_0}^{x_i} \tilde{n}_s^n \, dx, \quad (2.38)$$

where the second equality uses the definition of cell averages in the finite volume scheme (see (2.21)) and shows, that this numerical formula is exact. Another formula for updating $(E_x)_0$ to the time step $n$ is still needed. In the analytical case Gauss' law in combination with the density conservation equation (2.14) for the analytical flux (or cell values) $J_x \propto qnw_x$ can be shown to be equivalent to Faraday's law; in the numerical case this equivalency is shown using the discretized conservation equation and corresponding numerical flux $J_x \propto qF_n(\tilde{U}) \simeq qnw_x$ for the current density $J_x$. Taking the time derivative of (2.38) in conjunction with the discretized density update (2.23) leads to the expression

$$\frac{(E_x)_i^{n+1} - (E_x)_i^n}{\Delta t} + \frac{(E_x)_0^{n+1} - (E_x)_0^n}{\Delta t}$$
$$= \sum_s \frac{q_s}{\epsilon_0 \Delta t} \int_{t_n}^{t_{n+1}} [-(F_{n,s})_i + (F_{n,s})_0] \, dt. \quad (2.39)$$

The integration in time using Runge–Kutta methods is the same as used to solve (2.23). Faraday's law using fluxes in one spatial dimension is then given by

$$(E_x)_i^{n+1} = (E_x)_i^n - \sum_s \frac{q_s}{\epsilon_0} \int_{t_n}^{t_{n+1}} [F_n(\tilde{U})]_{i,s} \, dt, \quad (2.40)$$

and enables us to identify $J_x$ by comparison with the charge conservation equation ((2.14) multiplied by $q_s$)

$$(J_x)_i^{n+1/2} = \sum_s \frac{q_s}{\Delta t} \int_{t_n}^{t_{n+1}} [F_n(\tilde{U})]_{i,s} \, dt. \quad (2.41)$$

Note that the numerical flux also includes numerical diffusion and is directly related to changes in $\rho$. Due to this, other formulations for $J_x$ violate the charge conservation equation and can lead to numerical instabilities.

### 2.5.3. *Magnetic field evolution*

Because the fluid evolution influences the magnetic field only indirectly, the finite difference time-domain (FDTD) update for the magnetic field is unchanged from the previous SHARP code. For completeness we reproduce the formulae here (Shalaby *et al.* 2021)

$$(B_y)_i^{n+1/2} = (B_y)_i^{n-1/2} + \frac{\Delta t}{\Delta x}((E_z)_{i+1/2}^n - (E_z)_{i-1/2}^n), \tag{2.42}$$

$$(B_z)_i^{n+1/2} = (B_z)_i^{n-1/2} - \frac{\Delta t}{\Delta x}((E_z)_{i+1/2}^n - (E_y)_{i-1/2}^n). \tag{2.43}$$

Here, $B_x$ is constant in the 1D3V model because of the requirement $\nabla \cdot \boldsymbol{B} = 0$.

### 2.6. *Landau closure for fluid species*

The highest retained fluid moment, which is in our case the specific heat flux $\boldsymbol{Q}$, is not evolved in our set of equations. Instead, we need to estimate its value dynamically using an appropriate closure. The simple ideal gas closure sets $\boldsymbol{Q} = \boldsymbol{0}$, which, however, prevents the energy dissipation of plasma waves. One important mechanism of such a dissipation is the collisionless damping of electrostatic waves achieved through Landau damping. Landau damping is a microphysical kinetic wave–particle interaction, where particles resonate with the wave to exchange energy as a function of time. In essence, the resonant particles accelerate or decelerate to approach the wave's phase velocity, thereby picking up energy or releasing it, respectively. For Maxwellian phase-space distributions, there are more particles at velocities smaller than the phase velocity, which yields a net damping, i.e. energy loss of the wave (Boyd & Sanderson 2003).

Various attempts, e.g. by Hammett & Perkins (1990), were carried out to approximate the heat flux $\boldsymbol{Q}$ of an almost Maxwellian distributed plasma, such that the kinetic phenomenon of Landau damping is mimicked in the linearized fluid equations. Landau damping is a non-isotropic effect, which can be reflected in the fluid descriptions. Accounting for the gyrotropy of the system around the magnetic field, the double-adiabatic law with two adiabatic coefficients parallel and perpendicular to the magnetic field can be adopted, which might be extended using an appropriate heat flux vector (Snyder, Hammett & Dorland 1997; Hunana *et al.* 2019a,b). This is achieved by first decomposing the pressure tensor in a coordinate system that is aligned with the direction of the magnetic field $\boldsymbol{b} = \boldsymbol{B}/|\boldsymbol{B}|$, yielding $\boldsymbol{p} = p_\parallel \boldsymbol{bb} + p_\perp (\boldsymbol{1} - \boldsymbol{bb})$. The transport of parallel and perpendicular heat along the magnetic field then corresponds to the terms $Q_\parallel \boldsymbol{b}$ and $Q_\perp \boldsymbol{b}$, respectively. In this publication, our algorithm is restricted to pressures with only one common adiabatic coefficient for parallel and perpendicular pressures. We leave the possibility of implementing anisotropic double-adiabatic systems open for future extensions of our algorithm. Hence, we model only the scalar heat flux $Q = \boldsymbol{b} \cdot \boldsymbol{Q}$, which is a simplification of the double-adiabatic modelling from the aforementioned literature, i.e. $2Q/(\Gamma - 1) = Q_\parallel + 2Q_\perp$. In our model, we can assume an isotropized pressure tensor $\boldsymbol{p} = p\boldsymbol{1}$ using the adiabatic coefficient $\Gamma = 5/3$. To do this, we set $p = p_\parallel = p_\perp$ everywhere without explicitly modelling $Q_\perp$ and the corresponding perpendicular pressure equation. Instead, we choose $Q$ such that the linear Landau-damping rate of the isotropic system is comparable to that of an anisotropic electrostatic system with the same $p_\parallel$ (for details refer to comments below (2.46)). This is a simplifying assumption, which does not follow from the kinetic equations, because a rigorous treatment necessitates solving the perpendicular pressure equation. Furthermore, isotropization mainly results from collisions, while collisionless systems are rarely isotropic and Maxwellian as we assume.

Nevertheless, this is a convenient choice for damping waves with an isotropic background plasma when particle heating is negligible compared with their thermal energy. A more physically motivated anisotropic pressure tensor $\boldsymbol{p} = p\boldsymbol{e}_x\boldsymbol{e}_x$ is attained for $\Gamma = 3$, where $\boldsymbol{e}_x$ is the unit vector in the $x$-direction. This approximates the kinetic equations well when $p \simeq p_\parallel$. Since we model only components of $\boldsymbol{Q}$ parallel to $\boldsymbol{b}$, the heat flux enters into the pressure evolution equation (2.32) as

$$\frac{\partial p}{\partial t} \propto -\boldsymbol{\nabla} \cdot (\boldsymbol{b}Q). \tag{2.44}$$

In practice, we make the assumption of locally constant (or slowly varying) magnetic fields on top of the 1D3V geometry, that is $\boldsymbol{\nabla} \cdot (\boldsymbol{b}Q) \simeq \cos(\Theta)\partial_x Q$, where the angle $\Theta$ is measured between the background magnetic field $\boldsymbol{B}_0$ and the $x$-axis.

Here, we will introduce two different formulae for heat flux closures. The first and most popular collisionless electrostatic closure was proposed by Hammett & Perkins (1990). We refer to it as the $R_{32}$ closure throughout this paper, and it approximates the heat flux at a fixed $\Gamma = 3$, in Fourier space, by

$$\hat{Q} = -\mathrm{i}\,\mathrm{sign}(k)\frac{2}{\sqrt{\pi}}\sqrt{2\theta_0}cn_0 k_{\mathrm{B}}\frac{\hat{T}}{m} \equiv \hat{Q}_T. \tag{2.45}$$

Here, hats are used to denote quantities in Fourier space along the magnetic field line, i.e. $\hat{Q} = \mathcal{F}_\parallel(Q)$, and the subscript 0 refers to simulation box averages, that is $n_0 = \sum_{i=0}^{N_c} n_i/N_c$ is an average over all $N_c$ cells. Furthermore $k_{\mathrm{B}}$ is the Boltzmann constant, and $k_{\mathrm{B}}\hat{T} = (m\hat{p} - k_{\mathrm{B}}T_0\hat{n})/n_0$. Since the plasma average or equilibrium temperature evolves slowly as a function of time, we adjust the background temperature $T_0$ after every time step to synchronize it with the mean pressure, $k_{\mathrm{B}}T_0(t)/m = p_0(t)/n_0$, while the density conservation ensures that $n_0$ stays constant. Note also that $Q_0 = 0$. The dimensionless mass-normalized temperature is $\theta_0 = k_{\mathrm{B}}T_0/(mc^2)$.

A more recent approximation was proposed by Hunana *et al.* (2018), who restricts this closure to $\Gamma = 3$ only, for reasons mentioned already. We use an *ad hoc* formulation of their closure with a variable $\Gamma$, thereby allowing our simplified model to be used. They also introduce the nomenclature $R_{mn}$ adopted here, which is used to denote that the kinetic plasma response function $R$ is mimicked for this closure by a Padé approximant with polynomials $P_m/Q_n = R_{mn}$ of orders $m$ and $n$. We refer to their closure as $R_{31}$ and it approximates the heat flux, in Fourier space, by

$$\hat{Q} = \underbrace{\left(\frac{4}{4-\pi} - \Gamma\right)p_0\hat{w}}_{\hat{Q}_w} + \underbrace{\left(-\mathrm{i}\,\mathrm{sign}(k)\frac{\sqrt{2\pi\theta_0}}{4-\pi}cn_0\frac{k_{\mathrm{B}}\hat{T}}{m}\right)}_{\hat{Q}_T}. \tag{2.46}$$

The $R_{31}$ closure may be seen as a generalization of the $R_{32}$ closure with an additional degree of freedom in $\hat{w}$, which can mimic the kinetic damping more accurately over a broader spectrum. We choose this closure as our fiducial model because it requires only a relatively inexpensive local derivative to compute the additional term dependent on $\hat{w}$. This additional term effectively increases the speed of sound obtained from the non-electromagnetic fluid equations and allows us to retrieve the correct damping rate with our *ad hoc* assumption of a specific value of $\Gamma$, see Appendix C. This means, that adopting a value of $\Gamma$, e.g. $\Gamma = 5/3$ in our isotropic model, does not change the linear

dispersion relation associated with this closure. For $\Gamma = 3$, we retrieve the coefficient for $\hat{w}$ from the aforementioned literature $4/(4 - \pi) - 3 = (3\pi - 8)/(4 - \pi)$.

In only one spatial dimension, as assumed in our code, the global integration along a magnetic field line is approximated to be along the spatial direction, i.e. $\mathcal{F}_\parallel = \mathcal{F}_x$. An extension to multiple spatial dimensions with an anisotropic pressure tensor is not straightforward because, in this case, this approach can lead to spurious instabilities (Passot *et al.* 2014) and the integration would need to be carried out along magnetic field lines.

A kinetic code does not need global communication to accurately reproduce Landau damping, since each particle (or particle bin) tracks its own interaction with each wave mode as a function of time and accumulates this information in the particle velocity. However, after integrating out the individual particle velocities when building the evolution equations for the phase-space distribution function, i.e. (2.14)–(2.16), information about the individual particle–wave interaction is no longer collected. Because some information about this interaction is also contained in the wave, such non-local information can be used to approximate the gradient of the physical heat flux, i.e. a closure of the fluid moments that incorporates such missing information. This non-local information is approximated in (2.45) and (2.46), and is manifested by the term $i\,\mathrm{sign}(k)$ in Fourier space, which is also referred to as the Hilbert transform.

Numerically, we do not include the heat flux in the Riemann solver used to compute the fluid fluxes. Instead, we compute the spatial derivative of the heat flux $\nabla_\parallel \cdot Q$ separately. We use Strang splitting for the $w$-dependent part $Q_w$ and the temperature dependent part $Q_T$ to expand (2.20) into

$$U^{n+1/2} = \mathrm{e}^{(\Delta t/2)F}\, \mathrm{e}^{(\Delta t/2)Q_w}\, \mathrm{e}^{\Delta t Q_T}\, \mathrm{e}^{\Delta t S_{\mathrm{em}}}\, \mathrm{e}^{(\Delta t/2)Q_w}\, \mathrm{e}^{(\Delta t/2)F}\, U^{n-1/2} + O(\Delta t^3), \qquad (2.47)$$

such that only one non-global evaluation of $Q_T$ is needed. Using Heun's method together with the fast Fourier transform (FFT) the update formulae for the pressure with respect to operators $Q_w$ and $Q_T$ are, respectively,

$$p^{n+1}|_{Q_w} = \mathrm{e}^{\Delta t Q_w} p^n = p^n + \Delta t a_w p_0 \nabla_\parallel \cdot w, \qquad (2.48)$$

$$p^{n+1}|_{Q_T} = \mathrm{e}^{\Delta t Q_T} p^n = p^n + \Delta t \mathcal{F}_\parallel^{-1}\left[|k| a_T \left(1 + \frac{\Delta t}{2}|k| a_T\right)\hat{T}^n\right], \qquad (2.49)$$

where the derivative in Fourier space was obtained by multiplying with $ik$ and the inverse FFT is denoted by $\mathcal{F}^{-1}$. For the $R_{31}$ closure the coefficients are given by $a_w = 4/(4 - \pi)$ and $a_T = (4 - \pi)^{-1}(2\pi\theta_0)^{1/2}cn_0 k_B/m$, while for the $R_{32}$ closure these are given by $a_w = 0$ and $a_T = 2(2\theta_0/\pi)^{1/2}cn_0 k_B/m$. Both closures compute a term proportional to $\hat{T}$ (cf. (2.49))

$$ik\hat{Q} \propto -i\,\mathrm{sign}(k)ik a_T \hat{T} = |k| a_T \hat{T}. \qquad (2.50)$$

Computing this term naively using the FFT is expensive. This is why, in the following, we present local, semi-local and efficient global (Fourier transform-based) numerical approximations of the Landau closures, which we have implemented in the fluid-SHARP code.

### 2.6.1. *Local approximations of the Hilbert transform*

The phase shift between the wanted derivative $ik\hat{Q}$ and the input of $\hat{T}$ in (2.50) is exactly 0, while the amplitude is proportional to $|k|$. This is therefore a special case ($a = 1$)

of the fractional Riesz derivative $\partial^a/\partial|x|^a$ with Fourier representation

$$\mathcal{F}\left(\frac{\partial^a f(x)}{\partial|x|^a}\right) = -|k|^a \hat{f}(k), \tag{2.51}$$

where $a \in \mathbb{R}$. Note that all approximations mentioned here only introduce errors in the amplitude of $|k|$, but not in its phase. This makes them easier to integrate into simulations in comparison with approximations which are not designed to prevent phase errors, because large phase errors (between $\pi/2$ and $3\pi/2$) in any wave mode transform the damping term into an exponentially growing numerical instability. The local approximations make use of the fact, that the fractional Riesz derivative is local and cheap to evaluate for the special case $a = 2m$ with $m \in \mathbb{N}^0$, where it reproduces the usual derivative $\partial^{2m}/\partial|x|^{2m} = (-1)^{m+1}\partial^{2m}/\partial x^{2m}$. Wang *et al.* (2015) use $a = 0$, while Allmann-Rahn *et al.* (2018) and Ng *et al.* (2020) approximate the non-isotropic pressure tensors with $a = 2$. These approximations are scaled to a characteristic wavenumber $k_0$ at which the damping is expected to occur.

The choice of $a = 0$ means that the approximation is a scalar

$$ik\hat{Q} \propto |k_0|\hat{T}, \tag{2.52}$$

while the gradient-driven closures with $a = 2$ use

$$ik\hat{Q} \propto \frac{k^2}{|k_0|}\hat{T}. \tag{2.53}$$

The gradient-driven closures are equal to the FFT solution at two wavelengths, 0 and $k_0$, while the scalar closure is only exact at $k_0$, see figure 1. Since $ik\hat{Q}$ is not computed alongside the conserved fluxes in the Riemann solver, energy conservation is only preserved if the mean energy does not increase. To achieve this, the approximation for the derivative of the heat flux needs to vanish at wavenumber 0, which the scalar approximation does not fulfil.

Because fluid closures are only approximately mimicking kinetic Landau damping anyway, these local approximations to the fluid closures are useful to save computational cost. Furthermore they are easier to implement, especially when the full pressure tensor is computed. However, they may lead to misleading results in multiscale simulations, where multiple characteristic damping lengths are present and depend on the estimate of $k_0$. For example, Allmann-Rahn, Lautenbach & Grauer (2022) show a case where ion and electron heating intensities are switched qualitatively.

### 2.6.2. *Semi-local approximations of the Hilbert transform*

While the less accurate local approximations use an arbitrary value of $k_0$, the FFT is expensive and depends on periodic boundary conditions. Here, we aim to have a fallback algorithm as a compromise between both approaches.

A digital FIR filter can be designed to approximate the non-local effects by convolving the simulation data with adjacent auxiliary data points, where the filter length determines the maximum distance. For example, an asymmetric filter with an even number of entries is applied on an input $x$ using filter coefficients $b_j$, producing the output $y$

$$y_{i+0.5} = \sum_{j=-(N_f/2-0.5)}^{N_f/2-0.5} b_j x_{i+j+0.5}. \tag{2.54}$$
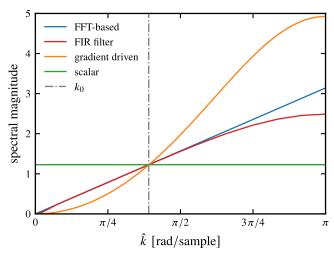
FIGURE 1. The magnitude of the frequency response, which is a quantification of how much the amplitude at a specific frequency is amplified or suppressed, of different approximations of the derivative of the Hilbert transform. Here, $\hat{k}$ is given in normalized frequencies (with regards to the Nyquist frequency), while the negative frequencies in the interval $[-\pi, 0]$ are not shown here due to the symmetric dependence of all plotted values on $|k|$. The FFT-based approach reproduces the correct, linear response. The scalar- and gradient-driven closures are given by (2.52) and (2.53), respectively, with the parameter $k_0$ marked as a grey vertical line. The FIR filter is described by (2.55).

A numerical derivative is then an asymmetrical filter with $N_f = 2$ and coefficients $b_{\pm 0.5} = \pm 1/\Delta x$, such that $y_{i+0.5} = (x_{i+1} - x_i)/\Delta x$. Figure 1 shows the magnitude of the frequency response. The gradient-driven case shows a quadratic $k^2$ dependence, which is suppressed for larger $k$. This is due to the relatively small uneven filter length of 7 used here; the filter length is an important parameter, since it influences the accuracy of the approximation. With a filter length corresponding to the simulation box size the results can converge to the FFT-based algorithm (i.e. the $k^2$ dependence is not suppressed at higher $k$), if the filter is designed appropriately. As noted previously, the local closures do not converge to $\partial/\partial|x|$. A correct convergence for approximating $\partial/\partial|x|$ is obtained through the high-order formulation by Ding, Li & Chen (2015). However, this filter violates energy conservation for smaller filter length and is thus not suitable for our case. Instead, we construct the filter by adopting a convolution of two sub-filters, each of which has an odd amount of asymmetric entries (termed a type IV filter) similar to the numerical derivative mentioned already. By design, their output has a vanishing mean, thereby guaranteeing energy conservation. A symmetric splitting into the sub-filters $\partial/\partial|x| = (\partial^{1/2}/\partial|x|^{1/2})^2$ is possible, however, its frequency response is not monotonic (and has visible ripples) for small filter lengths. This leads to the non-physical case that some waves at a particular wavenumber $k$ are damped less than their slightly larger scale waves at $k - \delta k$.

Instead, we opt to use the intuitive splitting of $\partial/\partial|x| = \partial/\partial x \mathcal{H}$ where the Hilbert-transform filter $\mathcal{H}$ is equivalent to $-i \, \text{sign}(k)$ in Fourier space. The filter $\mathcal{H}$ has coefficients $b_j = 1/(\pi j)$. We derive an equivalent formulation to (2.49), which is first order in time, by applying the derivative and Hilbert-transform filters successively, i.e.

$$p^{n+1} = p^n + \Delta t a_T \frac{\partial}{\partial x} \sum_{j=-(N_f/2-0.5)}^{N_f/2-0.5} \frac{1}{\pi j} T_{i+j+0.5}^n. \tag{2.55}$$

Note that the derivative is also computed by convolution and has a separate filter length corresponding to its spatial order. We opt to use the same spatial order as in the C-WENO reconstruction for the finite volume scheme.

Even for small Hilbert-transform filter lengths in comparison with the number of cells, e.g. $N_f/N_c = 0.04$ as shown in figure 1, this formulation dramatically improves the accuracy of multiscale problems in comparison with local approximations. Here, $N_f$ is critical for the accuracy at small wavenumbers $k$, while the spatial order of the derivative is critical for the accuracy at large $k$. Most importantly, this semi-local approach does not require setting of an arbitrary damping scale $k_0$ such as the local approximations mentioned before. The only parameter of this approach is the filter length, which should be chosen to be sufficiently large.

### 2.6.3. *Efficient FFT-based computation of the Hilbert transform*

Provided the plasma background is uniform and periodic, the most accurate while computationally most expensive results are achieved by computing the heat flux of the fluid in Fourier space. While the FFT is easy to compute on a single computer using standard numerical libraries, our code is parallelized using MPI and an efficient one-dimensional FFT is needed. The computation of the Fourier transform is expensive for two reasons:

(i) globally, each Fourier component needs to be informed about data from every other computational cell (which may be stored on a different processor); and

(ii) the Fourier transform is not easily parallelizable in one dimension, which precludes an efficient scalable Fourier algorithm.

This naturally limits the overall computational scalability of the fluid part of the code. Communication over multiple MPI processes is time consuming because of latency and finite bandwidth. For this reason, parallel FFT algorithms are prone to become a computational bottleneck. However, using non-blocking MPI routines to perform communication in the background can be used while the high computational load of the particles is carried out. Thus, in our case of a combined fluid and PIC algorithm, the communication required for an accurate FFT-based heat flux computation is comparatively computationally cheaper, even with relatively small numbers of PIC particles. Hence, in our case the FFT algorithm does not necessarily become a bottleneck for larger problems.

In order to distribute the computational load of the FFT, we employ a four-step algorithm in the first step of the computation (Bailey 1990; Takahashi & Kanada 2000), which extends the Cooley–Tukey algorithm (Cooley & Tukey 1965) for multiple processors. We shortly describe the algorithm for complex input data as found in the literature and afterwards adapt the parallel FFT for real input data in our implementation. The four-step algorithm interprets the complex data vector $x_j$ of length $N$ as a two-dimensional vector $x_j = x_{j_1,j_2}$ with lengths $n_1$ and $n_2$, respectively, and volume $n_1 n_2 = N$. The mapping $j = j_1 + j_2 n_1$ and $k = k_2 + k_1 n_2$ is inserted into the definition of the discrete Fourier transform, where $\Psi = \exp(-2\pi i)$

$$\hat{x}_k = \sum_{j=0}^{N-1} x_j \Psi^{jk/N}, \tag{2.56}$$

$$\hat{x}_{k_2,k_1} = \sum_{j_1=0}^{n_1-1} \sum_{j_2=0}^{n_2-1} x_{j_1,j2} \Psi^{j_2 k_2/n_2} \Psi^{j_1 k_2/N} \Psi^{j_1 k_1/n_1}. \tag{2.57}$$

This way, a complex-to-complex parallel FFT of length $N$ is distributed to $n_1$ local FFTs of length $n_2$, a multiplication by the twiddle factors $\Psi^{j_1 k_2/N}$ and finally $n_2$ FFTs of length $n_1$, with a communication intensive transpose in between. All-to-all communication takes place two times, in the first step – cyclically distributing $j$ to $j_1$ and $j_2$ – and for the transpose. A third all-to-all communication would be needed to properly sort the values in Fourier space. However, a scrambled output suffices for computing the heat flux. Furthermore, since often two FFTs, i.e. electrons and ions, need to be computed simultaneously, they can be computed on different nodes. This has the advantage that the second all-to-all communication for the transpose is not completely global resulting in reduced communication times.

Adapting this algorithm to a real-to-complex FFT, where, due to Hermitian symmetry, only values of $k \leq \lfloor N/2 \rfloor$ need to be computed, a large amount of computational and communicational savings can be realized. A real-to-complex parallel FFT of length $N$ is distributed to $n_1$ local real-to-complex FFTs of length $n_2$, a multiplication by the twiddle factors $\Psi^{j_1 k_2/N}$ and, now only, $\lfloor n_2/2 \rfloor + 1$ complex-to-complex FFTs of length $n_1$. Up to two of the latter FFTs can be replaced by real-to-complex FFTs, along the axis $k_2 = 0$ and, if $n_2$ is even, $k_2 = n_2/2$. A scrambled output is received, which, due to Hermitian symmetry, needs to be partially complex conjugated.

A key point in ensuring the efficiency of the parallel four-step algorithm consists in choosing large $n_1$ and $n_2$; $n_1 \simeq n_2 \simeq \sqrt{N}$ is the optimal choice for the distributed complex-to-complex FFT, the real-to-complex FFT should prefer $n_1 \simeq \lfloor n_2/2 \rfloor + 1 \simeq (\sqrt{2N+1} + 1)/2$. The computational scaling with $P$ processors and roughly optimally distributed $n_1$ and $n_2$ is akin to $O(N/P \log N)$, but degrades if $N$ is a prime number or, more generally, if $n_1$ or $n_2/2$ is smaller than the number of processors. This easily avoidable because $N$ is a free parameter, and so are $n_1$ and $n_2$. While this does not scale favourably in comparison with the $O(N/P)$ scaling that dominates the rest of the fluid code, still, the FFT is trivially independent of the numbers of particles per cell $N_{\mathrm{pc}}$. The PIC module on the other hand scales as $O(N_{\mathrm{pc}} N/P)$ and typical applications have $N_{\mathrm{pc}} \gtrsim 100$. In many applications the cost of the Fourier transform is, even with worse scaling, subdominant in comparison with the cost of the PIC part. In the remaining cases, local approximations, discussed above, are favourable.

### 2.7. *Current-coupled fluid-PIC algorithm*

The coupling in our code between various fluid and kinetic (PIC) species is achieved through a current-coupling scheme. Namely, both fluid and kinetic species contribute to the charge and current densities. The electromagnetic fields then evolve in response to the total contributions. The fields are staggered on a Yee mesh and are updated with the FDTD scheme. Subsequently, both fluid and kinetic species evolve in response to the new electromagnetic fields. That is our current-coupling scheme does not make any assumption on the velocity distribution of the species modelled using the kinetic description (Park *et al.* 1992).

The PIC species, using fifth-order spline interpolation, are deposited to specific points on the Yee grid for which the charge density is defined at full-time steps while the current density is defined at half-time steps, as discussed by Shalaby *et al.* (2017*b*, 2021). Table 1 gives an overview on the staggering of our implementation of the fluid-PIC method. We initialize the staggered quantities directly, with one exception: $(J_x)_f$, see (2.41), necessitates an integral over the flux from $t^0$ to $t^1$. We approximate the integral between $t^0$ and $t^{1/2}$ using an interpolation at $x_i$ of the cell centre values $J_x \simeq qnw|_{t^{1/2}}$, while the remaining part of the integral to $t^1$ is obtained through the fluxes again. If $E_x$ is updated using $J_x$, then $\rho$ does

|          | $x_i$        | $x_{i+1/2}$          |
|----------|--------------|----------------------|
| $t^n$    | $E_x$        | $E_y, E_z, \rho$     |
| $t^{n+1/2}$ | $B_y, B_z, J_x$ | $(J_y, J_z), U_f$ |

TABLE 1. The grid assignment and time staggering of the fluid-SHARP 1D3V code. Here, $U_f$ refers to the fluid state vector. Note that it is sufficient to compute either $\rho$ or $J_x$, but not both.

not need to be calculated and *vice versa*. Otherwise, another complication may be seen when obtaining $\rho$ from (2.5), which necessitates $U$ to be defined at full-time steps. While $U$ is formally defined only at half-time steps, we define $U^{n*} = \exp(\Delta t/2 \times \boldsymbol{F})U^{n-1/2}$ (i.e. the first part of the splitting in (2.20)), from which $\rho$ is obtained. Note that $\rho_s = q_s n_s$ stays constant when computing the Lorentz force and heat flux updates and therefore $\rho^n = \rho^{n*}$ is defined consistently, while bulk velocity and pressure are not well defined at full-time steps.

Our algorithm does not apply any approximations to the electrical field components or to Ohm's law, requiring electron time scales and motions to be fully resolved. Consequently, we apply the same algorithm to fluid electrons and protons. This is accomplished using the modular design of the fluid SHARP code, where each fluid species is represented by initializing a fluid code class. Each instance of this code class is initialized using the values of the mass and the charges of their respective particle species. The algorithms which define the evolution of each particle species are implemented as functions of the fluid class. This allows us to set up simulations with multiple species, all of which are evolved with the same numerical algorithms, with little effort.

In figure 2 the main loop of the fluid-PIC algorithm is presented schematically. It can be seen that the usual PIC-algorithm loop of electromagnetic update, interpolation to particle position, particle push and field deposition, is retrieved when no fluid species is initialized. On the other hand, without PIC particles, we retrieve a multispecies fluid plasma code. While our fluid-PIC algorithm can simulate an arbitrary mixture of species, it is most efficient if fluids are used for background species and particles for non-thermal particle distributions. Possibilities for task parallelization are shown in figure 2 by dashed lines, which allows maximizing of computation–communication overlap. The full main loop of our algorithm can be schematically described as follows (referencing the corresponding equations):

initialize quantities at corresponding grid points(table 1)
particle deposition
**while** $t < t_{\max}$:
    fluid deposition of $J_y, J_z$ (equations 2.36-2.37)
    fluid flux update by half$-$step (Section 2.4)
    fluid deposition of $J_x$ or $\rho$ (equation 2.41)
    electromagnetic update on Yee grid (Sections 2.5.2-2.5.3)
    $w$-dependent heat flux update by half step (equation 2.48)
    start $T$-dependent heat flux update (FFT, equation 2.49)
        fluid electromagnetic source update (Section 2.5.1)
        particle interpolation
        particle push
        particle deposition
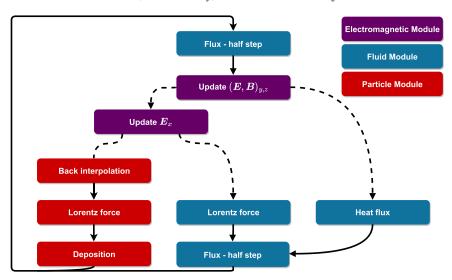    end $T$-dependent heat flux update (FFT, equation 2.49)

FIGURE 2. Schematic representation of the interaction of the different modules in the fluid-SHARP code. Red boxes belong to the particle class, violet boxes to the electromagnetic class and blue boxes to the fluid class. Dashed lines show branches which are task parallelizable, i.e. where non-blocking MPI communication can be used for overlapping communication and computation. The particle and fluid modules might be instantiated arbitrarily often, where each instance represents a species.

$w$-dependent heat flux update by half-step (equation 2.48)
fluid flux update by half−step (Section 2.4)
$t = t + \Delta t$

Our fluid implementation is included within the SHARP code, which uses a fifth-order spline function for deposition and back-interpolation for PIC species (Shalaby *et al.* 2017*b*, 2021). The PIC part of the code does not make use of filtering grid quantities and results in comparatively small numerical heating per time step, which (if present) would affect the reliability of the simulation results on long time scales (see § 5 in Shalaby *et al.* 2017*b*). This property is important because we are specifically interested in studying microphysical effects on long time scales with our fluid-PIC code. Noise generated by the PIC particles could influence the fluid through the electromagnetic coupling. The CFL condition keeps the propagation of this noise within a single cell during one time step, and the PIC noise at the next time step will be uncorrelated with this noise so that we do not expect a systematic numerical error emerging from this. Indeed, we have not yet observed a case where this leads to a numerical instability. It has also been observed, that the larger physical dissipation through Landau closures replaces the need for numerical dissipation completely (Passot *et al.* 2014).

Due to the modularity of our code, each part can be tested individually. These tests, ranging from the uncharged fluid solver to full fluid-PIC simulations, are shown in the next section.

## 3. Code validation tests

In this section, we present the results of various code tests. We start with two shock-tube tests in § 3.1, where only the fluid solver presented in § 2.4 without sources (electromagnetic module) is used. Next, we provide tests of the electromagnetic coupling

| Test | $x_0$ | $n_l$ | $w_l$ | $p_l$ | $n_r$ | $w_r$ | $p_r$ |
|------|-------|-------|-------|-------|-------|-------|-------|
| 1 | 0.3 | 1 | 0.75 | 1 | 0.125 | 0 | 0.1 |
| 2 | 0.8 | 1 | $-19.59745$ | 1000 | 1 | $-19.59745$ | 0.01 |

TABLE 2. Parameters adopted for the shock-tube tests described in § 3.1; $x_0$ divides the domain into two halves, where values to the left of $x_0$ ($x < x_0$) are initialized by the parameters with subscript $l$. Similarly, subscript $r$ indicates parameters to the right of $x_0$.

between ion and electron fluids, as described in § 2.5. The two-fluid model consists of an ion and electron fluid described by (2.14)–(2.16), coupled via Maxwell's equations (2.3) and (2.4). We show that our code is able to accurately capture all six branches of the two-fluid dispersion relation (§ 3.2). The Landau closures are tested for Langmuir wave damping of only one electrostatic electron fluid with a fixed ion background (§ 3.3) and, using the two-fluid model, for two interacting Alfvén waves generating a new, longitudinal wave along the magnetic field (§ 3.4). In § 3.5, we test the entire fluid-PIC code with a simulation of the gyrotropic CR streaming instability, where PIC CRs are streaming in a stationary electron–proton fluid background, utilizing two fluid and two PIC species coupled through Maxwell's equations. Finally, we demonstrate the successful parallelization strategy of our code by performing scaling tests in § 3.6.

### 3.1. *Shock tube*

As the fluid approximation will be primarily used for background plasmas without excessive gradients, the accuracy of resolving sharp discontinuities is of secondary importance in practical applications. Still, we stress test our implementation of the fluid equations without electromagnetic coupling to ensure its numerical robustness and to compare the numerical dispersion for different Riemann solvers. For the shock tests a numerical grid of 100 cells is used with a constant CFL number $C_{\text{cfl}} = 0.2$. The boundary conditions are transmissive and the initial conditions for the tests are given in table 2 with the adiabatic coefficient of $\Gamma = 1.4$. These test set-ups are the same as used by Toro (2009), where, unlike the tests performed here, a CFL number of $0.2 \times 0.95$ is used only in the first five steps and 0.95 afterwards. The units used for these non-electromagnetic tests are arbitrary units and do not coincide with the usual simulation units.

Test 1, as shown in figure 3(*a*), is a modified Sod shock-tube test. The sonic rarefaction wave on the left-hand side as well as the shock front on the right are well resolved without noticeable oscillations. The contact discontinuity in the middle introduces small oscillations in the density and is smeared out more than the shock front. While the Roe and HLLC solvers yield almost the same results, the HLLC solver is slightly better at resolving the sonic point at the head (to the left) of the sonic rarefaction wave, which the Roe solver can only resolve because an entropy fix is applied.

Figure 3(*b*) shows a test of a stationary contact discontinuity with a shock front of a high Mach number travelling to the right and a rarefaction wave to the left. It can be seen that, while the HLLC method introduces more oscillations, it is also better at resolving the contact discontinuity.

In low-density flows the Roe solver is not suitable because it is not robust without further modifications (Einfeldt *et al.* 1991), making the HLLC method slightly more robust while the Roe method is slightly less dispersive. We use the HLLC solver as our default, however, for most practical applications, both methods produce similar results.
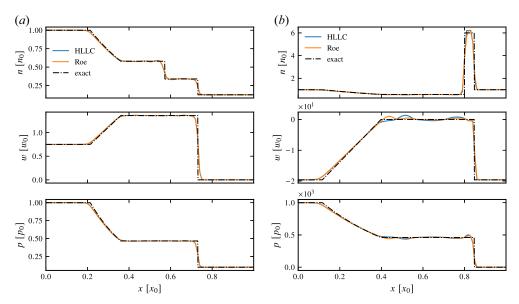
FIGURE 3. The 1D1V hydrodynamical shock-tube tests with initial conditions given in table 2. The simulations carried out with the HLLC and Roe Riemann solvers are compared with the exact solutions. Density, bulk velocity in the *x*-direction and pressure are plotted for each test.

A natural extension to the hydrodynamic shock tubes are the MHD shock tubes, which also test the evolution of shocks in the electromagnetic variables. A two-fluid model is not expected to exactly replicate the MHD shock tubes used to test MHD codes, because the characteristic waves are different for both system of equations. Finite volume two-fluid models have been used to replicate the MHD shock tubes with some success, even without informing the Riemann solver about the MHD characteristic wave velocity (Shumlak & Loverich 2003; Hakim *et al.* 2006). Because the Maxwell solver in our implementation uses the finite difference scheme, the most common choice for PIC codes, it is unable to capture electromagnetic shock tubes properly. Their relevance for two-fluid codes rarely extends beyond testing purposes, as physical shocks stretch over a length scale larger than $c/\omega_i \gg c/\omega_e$, which appears smooth in simulations resolving the electron skin depth. However, shock acceleration is not properly captured using the fluid-PIC algorithm at the shock interface. This is because efficient shock acceleration mechanisms are only experienced by the computational particles, but not the fluids. Injection prescriptions for CRs might be used to mitigate this (e.g. Pfrommer *et al.* 2017). We focus on cases where the electromagnetic quantities are smoothly varying, i.e. wave transport. The choice of the Riemann solver and its characteristic waves are less important for smooth waves, especially when employing a high-order interpolation routine. This is because different Riemann solvers should converge to the same results when the interface state is unambiguous, for example if $\tilde{U}_L = \tilde{U}_R$ (cf. (2.29)).

### 3.2. *Two-fluid dispersion relation*

To test the interplay of the fluid solver with the electromagnetic coupling, we perform a test where the linear waves of an ion–electron plasma are reproduced. For an ideal two-fluid plasma the dispersion relation can be solved for six different wave branches (Stix 1992). We show the solutions to the dispersion relation of a two-fluid plasma in figure 4 for a realistic mass ratio of $m_i = 1836 m_e$, $\beta_i = n k_B T_i / [B_0^2/(2\mu_0)] = 0.2$ in an isothermal
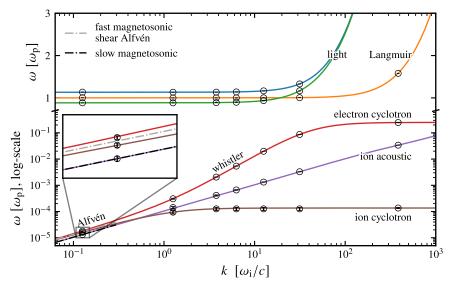
FIGURE 4. The six branches of the two-fluid dispersion relation are shown as lines, with two electrostatic wave branches (Langmuir and ion acoustic) as well as four electromagnetic wave branches, of which two are left- and two are right-hand circularly polarized (LCP and RCP). The lower RCP is referred to as the whistler or electron cyclotron branch and the lower LCP as ion cyclotron branch; for parallel propagation their phase velocities approach the Alfvén speed at small $k$. The upper RCP and LCP are modified light waves. We mark the six local extrema of the discrete Fourier-transformed fluid simulation outputs at each wavenumber with circles encapsulating the error bars extending over the frequency bin. For comparison, we plot the dispersion relation of the three MHD waves at scales larger than the ion inertial length, $1/k > c/\omega_i$.

plasma and $\Gamma = 3$. Here, $B_0$ is oriented along the $x$-axis and the Alfvén velocity is $v_A = B_0/(\mu_0 n_i m_i)^{1/2} = 5.83 \times 10^{-3} c$. Multiple simulations at different wavenumbers have been initialized that have all six wave modes simultaneously present and were run for a total time of $14 \times 2\pi/\min(\omega)$ ($20 \times 2\pi/\min(\omega)$ for the smallest scale), where $\omega$ denotes the wave frequencies, which are always completely real for an ideal fluid. Consequently, the waves should be undamped in the linear regime. Initial conditions for all of our fluid simulations of waves are obtained as eigenvectors (in $U$, $E$, $B$) while theoretical predictions of $\omega$ are obtained as eigenvalues using an extended algorithm based on the dispersion solver by Xie (2014), which can take into account the effects of both heat flux closures. We calculate the initial conditions to double precision, the machine precision of the simulation code. We normalize the amplitude of the eigenvectors by setting the maximum amplitude in any quantity to $10^{-4}$ for each wave mode, to suppress nonlinear effects. The resolution is $\Delta x = 0.1 c/\omega_e$ for all simulations. The box size for the intermediate scale is $L_x = 214.2 c/\omega_e$, covering waves with $k = n_w k_0$ of $n_w = 1, 3, 5, 10, 25$, where $k_0 = 2\pi/L_x$. The largest and smallest scales use box sizes of $L_x = 2142 c/\omega_e$ ($n_w = 1$) and $L_x = 21 c/\omega_e$ ($n_w = 3$), respectively. A Fourier analysis in time has been performed and the six largest local extrema are shown as encircled bars extending over a Fourier bin in figure 4. It can be seen, that the simulation results are in excellent agreement with the analytical results. In the Fourier analysis of the slowest two waves, the Fourier mode closest to the theoretical wave frequency is always observed. The largest error measured in this analysis occurs in the whistler branch for $k \sim 31 c/\omega_i$ with less than 0.5 %.
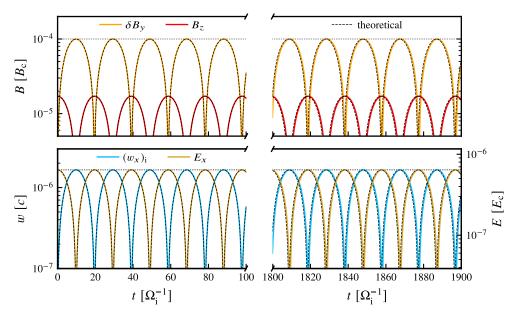
FIGURE 5. An oblique fast magnetosonic wave in a highly magnetized plasma. We plot the absolute of the perturbations of the magnetic field strengths in the top panel, as well as the ion bulk velocity and electric field along the domain in the bottom panel. The theoretical predictions are shown as black-dashed lines while the black-dotted lines indicate the amplitude. Electric and magnetic field strengths are expressed in code units, denoted by the subscript $c$.

Because our Riemann solver is not explicitly informed about MHD wave speeds, for which the coupling between fluids and electromagnetic fields is especially strong, one could naively expect large errors or numerical instabilities in the MHD limit. In order to test the fidelity of the coupling, we set up a wave with a wave speed well separated from the propagation speeds of the uncoupled fluid Riemann and electromagnetic solvers, i.e. $c_s \ll v_\phi = \omega/k \ll c$. We test a fast magnetosonic wave ($v_\phi = 0.03728c$ in two fluid vs. $v_{\text{fast}} = 0.03703c$ in MHD) at very low $\beta_{\parallel,\text{i}} = \beta_{\perp,\text{i}} = 0.02$ (corresponding to an oblique propagation angle of $\theta = 45°$ in the $B_y - B_x$ plane). The parameters of $k$, $\Delta x$, etc. are the same as for the parallelly propagating waves in the MHD limit in the previous test. While parallelly propagating electrostatic (electromagnetic) waves can be described using 1D1V (1D2V) models, oblique propagation requires the 1D3V model. In figure 5 the time evolution of two representative quantities along and perpendicular to the box size direction are shown. The time evolution of the different quantities are taken from the first cell in the simulation box. The fast magnetosonic wave is captured well, short and long term, without introducing numerical instabilities at low dissipation. The simulated wave velocity is 0.027 % slower than the theoretical prediction. In conclusion, the current-coupled fluid and electromagnetic solvers numerically approximate the analytical dispersion relation with high fidelity.

### 3.3. *Langmuir wave damping*

The electrostatic wave modes are directly subject to linear Landau damping, and thus present a good test for the heat flux closures. To test this, we initialize standing Langmuir waves in an electron plasma with immobile ions. We use the same grid layouts as in table 1 of Shalaby *et al.* (2017*b*), supplemented with fluid simulations run at $k/k_{\text{D}} \in \{0.1, 0.2, 0.3\}$
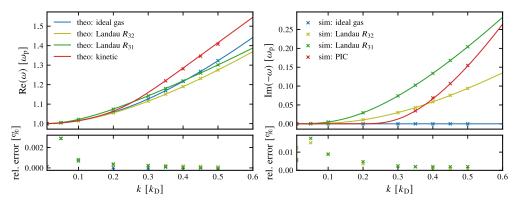
FIGURE 6. The linear dispersion relations of a Langmuir wave with immobile ions. Shown are, on the left-hand side, the real frequency components and, on the right-hand side, the negative imaginary frequency components (which are responsible for damping). The crosses present data points obtained from simulations with the respective closure while the theoretical result is shown with a solid line. The relative error between simulation and theoretical results $(\omega^{\text{sim}} - \omega^{\text{theor}})/\omega^{\text{theor}}$ is shown in the lower panels. For reference, the red crosses display the data points as given in table 1 of Shalaby *et al.* (2017*b*).

with a resolution of $\lambda/\Delta x = 68$ cells per wavelength and a domain size of length $L = 10\lambda$ wavelengths. The wavenumber associated with the Debye length is the ratio of plasma frequency to thermal velocity, i.e. $k_{\text{D}} = \omega_p/\theta^{1/2}c$. The amplitude of the wave is chosen, such that the density fluctuation to background ratio is fixed to $\delta n/n_0 = 10^{-3}$.

In order to find the numerical dispersion relation we perform curve fitting with the Powell algorithm on the time series for times up to $80\omega_p^{-1}$, while the simulations at $k/k_{\text{D}} = 0.01$ and $0.05$ with small damping are analysed up to $240\omega_p^{-1}$. The computation of the heat fluxes for the $R_{31}$ and $R_{32}$ closures is performed using the FFT-based method. The results are shown in figure 6, where the ideal gas closure and the kinetic results are also depicted for reference (using $\Gamma = 3$).

Generally, it can be seen that, at small scales, the closures show larger deviations from each other, which is also where the fluid description starts breaking down naturally as the particle distribution is not in equilibrium. At larger scales, the various descriptions of Landau damping converge and approach zero. The numerical relative error of the fluid code is small and stays below 0.003 % for real frequencies and below 0.02 % for decay rates in this set-up. The simulation at $k/k_{\text{D}} = 0.05$ performs worse than the one at $k/k_{\text{D}} = 0.1$ due to the significantly lower resolution. The error in $\omega$ decreases at second order with increasing spatial resolution, as shown in Appendix B.

### 3.4. *Interacting Alfvén waves*

A single Alfvén wave is purely transversal and not directly affected by Landau damping. However, two or more Alfvén waves drive a longitudinal electrostatic wave, which is susceptible to Landau damping, see figure 7. This leads to particle heating as a result of the collisionless damping of the Alfvén wave, also known as nonlinear Landau damping (Lee & Völk 1973).

Restricting ourselves to a set-up of pairwise interacting waves, we can identify two distinct cases. In the first case counter-propagating waves are interacting. In consequence, both waves damp, lose energy to the longitudinal wave and subsequently heat the particles. In the second case the waves are co-propagating. Here, the wave with the smaller
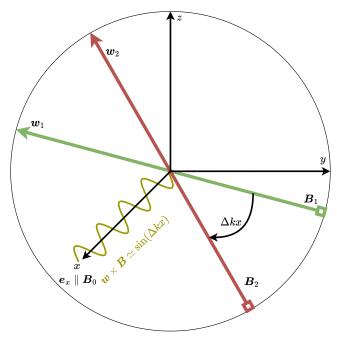
FIGURE 7. Two different Alfvén waves, with magnetic and velocity vectors $B_1, B_2$ and $w_1, w_2$, propagate transversally along the $x$-axis, where the electromagnetic vectors rotate (counter-)clockwise around it. Because of their phase difference $\Delta kx$ the overall Lorentz force $(w_1 + w_2) \times (B_1 + B_2)$ in the $x$-direction is non-zero, thereby generating the longitudinal wave shown in dark yellow.

wavelength will not only transfer energy to the particles, but also to the other Alfvén wave. Lee & Völk (1973) describe this mechanism in detail and formulate the following coupled set of differential equations while adopting a measure for the magnetic energy of a wave, $I_j = |B_j|^2$, where $j \in \{1, 2\}$:

$$\frac{\mathrm{d}}{\mathrm{dt}} I_j = 2\Gamma_j I_j. \tag{3.1}$$

The coupling between the differential equations is implicit because the damping coefficient has the dependency $\Gamma_1 \propto I_2$. For the counter-propagating case with an isothermal ion–electron plasma in the high beta limit $\beta_i = 2\mu_0 n_i k_B T_i / B_0^2 = 2 \gg 1$, where $B_0$ is the background magnetic field strength, the damping rate $\Gamma_j$ is approximately equal for both wave polarizations with similar frequencies $\omega_j$ and may be approximated by (Holcomb 2019)

$$\Gamma_1 = -\frac{\sqrt{\pi}}{16} \frac{I_2}{B_0^2} \sqrt{\beta_i} \omega_1. \tag{3.2}$$

Note that $\Gamma_2$ is found by substituting the subscripts $1 \to 2$ and $2 \to 1$. This prediction is using kinetic physics and also includes damping effects due to modulation in $B_\perp$ (see figure 5), which can electromagnetically heat or even trap particles, analogous to Landau damping in the electrostatic case. This is not captured in the Landau fluid approximation. Therefore, we do not expect our analytical and simulated damping rates to exactly match. However, they provide a good insight into whether wave modes are qualitatively correctly
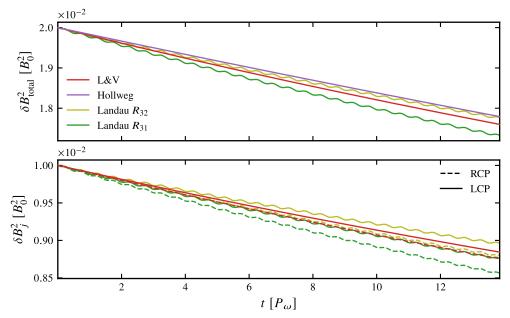
FIGURE 8. Time evolution of the magnetic energy of a linearly polarized Alfvén wave in our fluid simulations with Landau damping. Time is measured in units of the period of the mean wave frequencies $P_\omega = 4\pi(\omega_1 + \omega_2)^{-1}$. Analytical predictions for the damping rate are taken from Lee & Völk (1973, labelled L&V) and Hollweg (1971). The fluid simulations are presented with the different heat flux closures $R_{31}$ and $R_{32}$. We compare the time evolution of the total magnetic wave energy (*a*) and the magnetic wave energy of the different polarization states (*b*). The RCP wave has a higher phase velocity and loses energy more quickly in comparison with the LCP wave.

captured. Another prediction by Hollweg (1971) uses the fluid picture to derive the amplitude of the secondary, electrostatic wave, which is then damped according to kinetic prescriptions. This prediction agrees with our model. However, this analysis does not differentiate different wave types and therefore does not make individual predictions about interacting waves. In the case considered in the following, the damping rates are coincidentally similar.

In figure 8 we show simulations of a linearly polarized Alfvén wave, which consists of two counter-propagating waves of equal amplitude. The pure fluid simulations are shown with a box size of $L = 252c/\omega_i$ and wavelengths $\lambda = L/3$. Right- and left-hand circularly polarized waves are initialized with phase velocities $\omega_{RCP}/k = 0.0342$ and $\omega_{LCP}/k = 0.0318$ with a perpendicular magnetic field amplitude of $\delta B = 0.1B_0$. A reduced mass ratio of $m_i/m_e = 100$ is adopted here.

Our simulations are carried out with the different heat flux closures $R_{32}$ and $R_{31}$, as shown in figure 8. Both closures reproduce the theoretical predictions quite well. A PIC simulation with similar parameters has been shown in figure 6.4 by Holcomb (2019), which reproduces half of the predicted damping rate until $t \sim 2P_\omega$ and shows a quenching of the damping rate afterwards. In comparison with kinetic simulations, there is no saturation of the Landau-damping effect in fluids. This is because the distribution of the fluid particles is always assumed to be roughly Maxwellian and resonant particles are not depleted as a function of time. Hence, Landau fluids are implicitly assumed to have small thermalization time scale in comparison with the damping time scale. On the other hand, PIC simulations

are plagued by Poisson noise and an insufficient resolution of velocity space might lead to a reduced Landau damping rate.

### 3.5. *Gyrotropic CR streaming instability*

To test the entire code, we run CR streaming instability simulations, where electron and ion CRs are modelled with the PIC method and the background electron and ion plasmas are modelled as fluids. The initial CR momentum distribution for ions (electrons) is assumed to be a gyrotropic distribution with a non-vanishing (zero) pitch angle, while both CR electrons and ions are assumed to drift at the same velocity $v_{\mathrm{dr}}$. Namely, the phase space distributions for the electron and ion CR species $s \in \{\mathrm{e, i}\}$ are given by (Shalaby *et al.* 2021, 2023)

$$f_{\mathrm{cr},s}(\boldsymbol{x}, \boldsymbol{u}) = \frac{n_{\mathrm{cr},s}}{2\pi u_{\perp}} \delta(u_{\parallel} - \gamma_s v_{\mathrm{dr}}) \delta(u_{\perp} - \gamma_s v_{\perp,s}), \tag{3.3}$$

where $\gamma_s = (1 - v_{\mathrm{dr}}^2/c^2 - v_{\perp,s}^2/c^2)^{-1/2}$ is the Lorentz factor and $v_{\perp,s}$ is the perpendicular component of the CR velocity. We choose $v_{\perp,\mathrm{e}} = 0$ and $v_{\perp,\mathrm{i}} = 13.1 v_{\mathrm{A}}$, where the ion Alfvén velocity is given by $v_{\mathrm{A}} = B_0/(\mu_0 n_{\mathrm{i}} m_{\mathrm{i}})^{1/2} = 0.01c$ with the background magnetic field pointing along the spatial direction, and $v_{\mathrm{dr}}$ of $5 v_{\mathrm{A}}$ resulting in a pitch angle for the ions of $\tan^{-1}(v_{\perp,\mathrm{i}}/v_{\mathrm{dr}}) = 69.1°$. The thermal background species are isothermal with the temperatures $k_{\mathrm{B}} T/(mc^2) = 10^{-4}$ and a mass ratio $m_{\mathrm{i}}/m_{\mathrm{e}} = 1836$. We use a periodic box of length $L_x = 10\,971.5 c/\omega_p$ and resolution $\Delta x = 0.1 c/\omega_p$. The CR to background number density ratio $\alpha = n_{\mathrm{cr,i}}/n_{\mathrm{i}} = 0.01$.

We run two simulations where the background plasmas are modelled as fluids. The first one uses an ideal gas closure without accounting for Landau damping (FPIC ideal gas) while we include the heat flux source term in the second simulation to mimic the impact of linear Landau damping using the $R_{31}$ closure of (2.46) (FPIC Landau $R_{31}$). We compare these two fluid-PIC simulations against PIC simulations where both CRs and background plasmas are modelled as PIC species. The number of CR ions per cell is $N_{\mathrm{pc}} = 25(75)$ and we call this simulation 'PIC normal (high) $N_{\mathrm{pc}}$' (Shalaby *et al.* 2021). Like the 'PIC normal $N_{\mathrm{pc}}$' simulation, the fluid-PIC simulations also use 25 particles per cell for modelling CRs.

Growth rates of the instability in the linear regime can be computed from the linear cold background plasma dispersion relation (Holcomb & Spitkovsky 2019; Shalaby *et al.* 2022)

$$0 = 1 - \frac{k^2 c^2}{\omega^2} + \frac{\omega_{\mathrm{i}}^2}{\omega(-\omega \pm \Omega_{\mathrm{i,0}})} + \frac{\omega_{\mathrm{e}}^2}{\omega(-\omega \pm \Omega_{\mathrm{e,0}})} + \frac{\alpha \omega_{\mathrm{e}}^2}{\gamma_{\mathrm{e}} \omega^2} \left( \frac{\omega - k v_{\mathrm{dr}}}{k v_{\mathrm{dr}} - \omega \pm \Omega_{\mathrm{e,0}}} \right)$$
$$+ \frac{\alpha \omega_{\mathrm{i}}^2}{\gamma_{\mathrm{i}} \omega^2} \left( \frac{\omega - k v_{\mathrm{dr}}}{k v_{\mathrm{dr}} - \omega \pm \Omega_{\mathrm{i}}} - \frac{v_{\perp}^2/c^2 (k^2 c^2 - \omega^2)}{2 (k v_{\mathrm{dr}} - \omega \pm \Omega_{\mathrm{i}})^2} \right). \tag{3.4}$$

The non-relativistic and relativistic cyclotron frequencies of each species are given by $\Omega_{s,0} = q_s B_0/m_s$ and $\Omega_s = \Omega_{s,0}/\gamma_s$, respectively. The wavelength of the most unstable wave mode at the gyroscale is $\lambda_g = 2\pi(v_{\mathrm{dr}} - v_{\mathrm{A}})/\Omega_{\mathrm{i}}$, which is properly captured in our set-up using a box size of $L_x \sim 10.15 \lambda_g$.

We show the amplification of the perpendicular magnetic field components as a function of time for this unstable set-up in figure 9 for various simulations. It shows that the noise level of the fluid-PIC simulations is orders of magnitude lower in comparison with the 'PIC normal $N_{\mathrm{pc}}$' resolution, even though the number of CR particles per cell is the same. Especially up to the saturation point ($t\Omega_{\mathrm{i}} \sim 10$) the fluid-PIC simulation compares more favourably with the PIC results with lower noise than with the PIC simulation with fewer $N_{\mathrm{pc}}$.
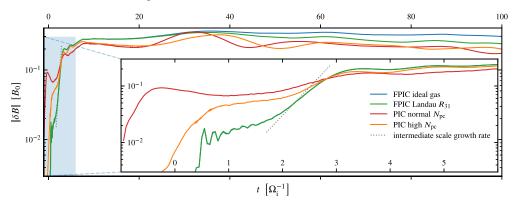
FIGURE 9. Growth of the perpendicular magnetic field as a function of time for a gyrotropic CR streaming set-up. The maximum growth rate expected from the linear dispersion relation at intermediate scales is $\Gamma_{\text{inter}} = 2.299\Omega_i$ and shown in dashed grey. Because of the different initial seed populations for the particle species, the onset of the instabilities is not expected to happen at the same simulation time. Hence, we choose an arbitrary $t = 0$ so that the different simulated growth phases roughly coincide.

After saturation, i.e. when Alfvén waves at many scales have built up and their interaction has created an electrostatic field, these waves start to lose some energy to Landau damping of the electrostatic waves (see § 3.4). At that point, the Landau closure becomes relevant. Qualitatively the ideal gas closure has no efficient mechanism for dissipating such electrostatic waves, resulting in a prolonged growth period leading to saturation at higher values at the cascading and intermediate scales. Utilization of a Landau closure leads to some damping, albeit it is quantitatively smaller than in the PIC simulations. While figure 6 indicates faster damping for the Landau closures in comparison with the kinetic results in the electron electrostatic branches, damping in the ion-acoustic branch might be underestimated in the Landau closures. We have compared the expected damping between kinetic and Landau fluid in the ion-acoustic branch for multiple wavenumbers, which confirmed that this is a likely scenario. The accuracy of this approximation is not the same at all scales, which can be seen in figure 10, where the magnetic field amplifications at various ranges of scales are compared. Especially at the highly Landau-damped scales, differences between fluid-PIC and PIC emerge. At ion gyro-scales, where most of the magnetic energy is stored at saturation, there is a good agreement over the entire time period. Exponential growth at every scale is also in good agreement between PIC and fluid-PIC simulations at all scales. The initial exponential growth can also be compared with the expected growth rates from the linear dispersion relation. The growth rates of the two local maxima are plotted alongside the simulated data, one at the intermediate scales around $ck = 4.91\omega_i$ and one at the gyro-scale at $ck = 0.38\omega_i$. The intermediate scale starts an inverse cascade to larger scales almost immediately, which causes a reduced growth rate in comparison with the expectation from linear theory. By contrast, the gyro-scale instability follows linear expectations to very good approximation.

While our fluid-PIC and PIC results are promisingly similar, differences after the saturation level might be attributed to multiple reasons. First, the Landau closures do not exactly reproduce the correct damping, and therefore will deviate quantitatively. Second, due to the high electron temperature chosen, relativistic effects might occur in PIC, but not in the non-relativistic fluid that we assumed for the background plasma. Third, the
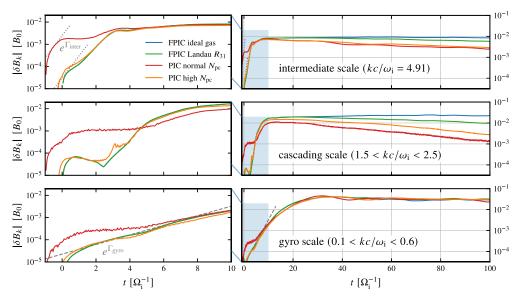
FIGURE 10. Growth of the perpendicular magnetic field as a function of time at different scales for a gyrotropic CR streaming set-up. We show mean values of the fields that are averaged over a range of wave vectors $k$, as indicated in the legends. The maximum growth rates at the gyro-scale and the intermediate scale are given by $\Gamma_{gyro} = 0.498\Omega_i$ and $\Gamma_{inter} = 2.299\Omega_i$, and indicated by the grey dotted and dashed lines, respectively. At wavenumbers corresponding to cascading scales, there is no instability expected according to the linear dispersion relation, and wave growth solely arises as a result of cascading from other (unstable) scales.

PIC method might exhibit more numerical dissipation at the given $N_{pc}$ in comparison with the fluid method. However, figure 10 seems to indicate numerical convergence at the intermediate and gyro-scale.

Even though our simulations were run at unrealistically high $\alpha$, the background particles did not deviate significantly from the Maxwellian distribution at the end of the simulation time. The pressure anisotropy measured from the PIC thermal particles is below 2 %. This indicates that an isotropic fluid description for background species is a valid approach for this set-up, especially for smaller, more realistic values of $\alpha$.

### 3.6. *Computational scaling*

We show the strong scaling properties of our fluid-PIC code in figure 11. The tests were run on Intel Cascade 9242 processors with 96 processors per node at the HLRN Emmy cluster. Simulations with 3000 processors or more typically cause severe bottlenecks due to the latency and/or the finite bandwidth of input/output operations. For this number of processors the Fourier-based closures are roughly 20 % more costly in comparison with the ideal gas closures. This is in stark contrast to pure PIC simulations, which scale with the inverse ratio of CR-to-background density $\alpha^{-1}$, consequently the fluid-PIC algorithm leads to a speed up of a factor of 100 for the simulation performed in § 3.5, which adopted unrealistically large $\alpha$.

The bottleneck in the communication procedure of our implementation is currently the 'Ialltoallv' MPI routine, which is not optimized for hierarchical architecture networks as of now. Further optimizations to this might provide fruitful in increasing the code's scalability further if necessary.
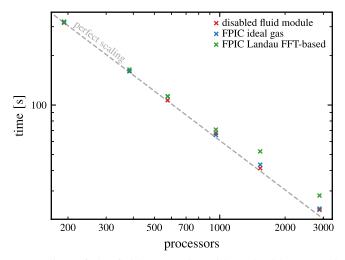
FIGURE 11. Strong scaling of the fluid-PIC code, with and without Fourier-based Landau closures. Shown is the wall-clock time needed to simulate 1250 time integration steps with 180 000 cells at 1000 particles per cell at a varying number of processors. We show the perfect strong scaling that is proportional to the inverse number of processors as the grey dashed line for reference. For the disabled fluid module no background plasma was initialized and only CRs are initialized, showing that the bulk of the computational work is performed by the PIC routines.

The fluid-PIC simulations in § 3.5 used only $N_{pc} = 25$ and seem to be sufficiently resolved. For such a low particle number, the FFT is the bottleneck for scalability because the overlap of communication and computation is small, i.e. we measure a 260 % increase in time with 2880 processors, while at 192 processors the increase is below 20 %. This indicates that scalability of fluid-only simulations is dominated quickly by the FFT, while the cost is almost negligible for fluid-PIC simulations. Still, simulations with only a few particles per cell are computationally inexpensive so that there is no reason for performing such a simulation on thousands of processors. Furthermore, the example of a mono-energetic cold CR beam is not very demanding regarding the phase-space resolution. More realistic scenarios include power-law distributions for the CR population as well as larger spatial density inhomogeneities, both resulting in an increased requirement for the number of particles per cell in order to accurately resolve the velocity phase-space distribution along the entire spatial domain.

## 4. Conclusion

In this paper, we introduce a new technique termed fluid-PIC, which uses Maxwell's equations to self-consistently couple the PIC method to the fluid equations. This technique is particularly aimed at simulating energetic particles like CRs interacting with a thermal plasma. This enables us to resolve effects on electron time and length scales and to emulate Landau damping in the fluid by incorporating appropriate closures for the divergence of the heat flux. The underlying building blocks of our implementation are the SHARP 1D3V PIC code extended by a newly developed fluid module and the overall algorithm is second-order accurate in space and time. While an ideal fluid does not exhibit Landau damping, we have implemented two different Landau fluid closures and studied their performance. Here we summarize our main findings:

- We developed a multi-species fluid code that is coupled to explicit PIC algorithm. In order to couple multi-fluid equations to Maxwell's equations, very often implicit

and semi-implicit methods have been used for stability reasons. However, the resulting interdependency between all fluids complicates their coupling to explicit PIC methods. To ensure numerical stability, Riemann solvers that provide some numerical diffusion are used. However, we demonstrate that the level of numerical diffusivity needs to be small enough so that it does not numerically damp physical small-amplitude plasma waves or quench plasma instabilities. We confirm the numerical stability and small dissipation of our implementation by employing a diverse range of test set-ups that test the coupling between the fluid and electromagnetic modules. Most importantly, our new fluid-PIC code fully resolves the electron time scales, precluding the need to adopt any simplifying assumptions to the electrical field components or to Ohm's law. This enables the versatility of our implementation, allowing us to instantiate an arbitrary number of species, which can be modelled individually either as a fluid or as particles.

- We compare various Landau fluid closures and demonstrate that local closures only produce reliable results close to a characteristic scale while they are prone to fail in multi-scale problems. By contrast, semi-local spatial filters or global (Fourier-based) methods to estimate Landau fluid closures produce reliable results for a large range of scales. Most importantly, we demonstrate that the inclusion of communication intensive (Fourier-based) fluid closures only have a minimal impact on our code performance (through the usage of non-blocking background communication) because the majority of the computational workload is taken up by the much more cost-intensive PIC module. This enables us to make use of the more accurate Fourier-based Landau closure for the fluid instead of relying on local approximations only.

- In numerical tests, our implementation of the multi-species fluid module showed excellent agreement with theoretical frequencies and damping rates of Langmuir waves, oscillation frequencies of various two-fluid wave modes, as well as the nonlinear Landau damping of Alfvén waves.

- First simulations of the CR streaming instability with our combined fluid-PIC code provide very good agreement with the results of pure PIC simulations, especially for the growth rates and saturation levels of the gyro-scale and intermediate-scale instabilities. This success is achieved at a substantially lower Poisson noise of the background plasma at the same number of computational CR particles per cell. Most importantly, the numerical cost of the fluid-PIC simulation is reduced by the CR-to-background number density ratio. However, we find that the late-time behaviour of the CR streaming instability differs for our fluid-PIC and PIC simulations. More work is needed to understand the reason for this, which could be either resulting from (i) numerical damping due to Poisson noise resulting from the finite number of PIC particles, (ii) missing relativistic (electron) effects in our non-relativistic fluid dynamics or (iii) missing physics in our fluid closures that may be underestimating other relevant collisionless wave damping processes.

Three possible future extensions of the algorithm are left open here. (i) Extending the fluid formulation with a full pressure tensor, (ii) extending the code to two or three spatial dimensions and (iii) the inclusion of direct interaction terms between the various fluids to explicitly incorporate scattering processes such as ion–neutral damping. The novel fluid-PIC framework greatly extends the computationally limited parameter space accessible to pure PIC methods whilst not compromising on some of the most important microphysical plasma effects. This opens up many possibilities for studying CR physics in physically relevant parameter regimes, such as the growth and saturation of

the CR streaming instability in different environments, and including the effect of partial ionization, ion–neutral damping and inhomogeneities of the background plasma.

## Declaration of interest

The authors report no conflict of interest.

## Data availability

The data underlying this article will be shared on reasonable request to the corresponding author.

## Appendix A. The C-WENO coefficients

We list all coefficients needed to implement the C-WENO reconstruction in this section. Because our reconstruction procedure is applied component-wise to each of the primitive variables, we assume for this appendix that we are reconstructing a single quantity $u$. The smoothness indicator for the low-order polynomials are given by Jiang & Shu (1996)

$$\mathrm{IS}[P_L] = \tfrac{13}{12}(u_{i-2} - 2u_{i-1} + u_i)^2 + \tfrac{1}{4}(u_{i-2} - 4u_{i-1} + 3u_i)^2, \tag{A1}$$

$$\mathrm{IS}[P_C] = \tfrac{13}{12}(u_{i-1} - 2u_i + u_{i+1})^2 + \tfrac{1}{4}(u_{i+1} - u_{i-1})^2, \tag{A2}$$

$$\mathrm{IS}[P_R] = \tfrac{13}{12}(u_i - 2u_{i+1} + u_{i+2})^2 + \tfrac{1}{4}(3u_i - 4u_{i+1} + u_{i+2})^2, \tag{A3}$$

while four auxiliary variables are defined

$$D_1 = \frac{(6w_0 - 1)(u_{i-2} + u_{i+2}) - 2(18w_0 - 1)(u_{i-1} - u_{i+1})}{48w_0}, \tag{A4}$$

$$D_2 = \frac{(2w_0 - 3)(u_{i-2} + u_{i+2}) - 2(2w_0 + 9)u_i + 12(u_{i-1} + u_{i+1})}{16w_0}, \tag{A5}$$

$$D_3 = \frac{-u_{i-2} + 2(u_{i-1} - u_{i+1}) + u_{i+2}}{12w_0}, \tag{A6}$$

$$D_4 = \frac{u_{i-2} - 4u_{i-1} + 6u_i - 4u_{i+1} + u_{i+2}}{24w_0}, \tag{A7}$$

to define the smoothness indicator for the $P_0$ polynomial

$$\mathrm{IS}[P_0] = D_1^2 + \tfrac{13}{3}D_2^2 + \tfrac{3129}{80}D_3^2 + \tfrac{87617}{140}D_4^2 + \tfrac{1}{2}D_3 D_1 + \tfrac{21}{5}D_2 D_4. \tag{A8}$$

The overall smoothness indicator is given by Cravero *et al.* (2018*a*)

$$\tau = |\mathrm{IS}[P_L] - \mathrm{IS}[P_R]|. \tag{A9}$$

The low-order polynomials are evaluated at the left-hand interface of a given cell via

$$P_L(x_{i-1/2}) = \tfrac{1}{6}(-u_{i-2} + 5u_{i-1} + 2u_i), \tag{A10}$$

$$P_C(x_{i-1/2}) = \tfrac{1}{6}(2u_{i-1} + 5u_i - u_{i+1}), \tag{A11}$$

$$P_R(x_{i-1/2}) = \tfrac{1}{6}(11u_i - 7u_{i+1} + 2u_{i+2}), \tag{A12}$$

while they evaluate to

$$P_L(x_{i+1/2}) = \tfrac{1}{6}(2u_{i-2} - 7u_{i-1} + 11u_i), \tag{A13}$$

$$P_C(x_{i+1/2}) = \tfrac{1}{6}(-u_{i-1} + 5u_i + 2u_{i+1}), \tag{A14}$$

$$P_R(x_{i+1/2}) = \tfrac{1}{6}(2u_i + 5u_{i+1} - u_{i+2}), \tag{A15}$$

at the right-hand interface. The optimal polynomial evaluates to

$$P_{\text{opt}}(x_{i-1/2}) = \tfrac{1}{60}(-3u_{i-2} + 27u_{i-1} + 47u_i - 13u_{i+1} + 7u_{i+2})$$
$$= \tfrac{1}{10}[3P_L(x_{i-1/2}) + 6P_C(x_{i-1/2}) + P_R(x_{i-1/2})], \tag{A16}$$

$$P_{\text{opt}}(x_{i+1/2}) = \tfrac{1}{10}[P_L(x_{i+1/2}) + 6P_C(x_{i+1/2}) + 3P_R(x_{i+1/2})], \tag{A17}$$

at both interfaces of the cell. The interface values of $P_0$ can be derived from (2.24).

## Appendix B. Convergence order

In order to numerically prove a second-order scaling of the plasma frequency for the different heat flux closures, the linear dispersion of the Langmuir wave set-up described in § 3.3 is simulated at different resolutions of $\lambda/\Delta x$. We concentrate here on the convergence of a wave with wavenumber $k/k_{\text{D}} = 0.05$. The results are shown in figure 12 and demonstrate a very good match with the predicted errors assuming a second-order convergence. At first sight, the Landau closures do not seem to scale ideally for higher resolutions. However, this is the result of physical plasma heating due to wave damping in our set-up leading to a nonlinear increase in the expected plasma frequency. The accuracy of the spatial integration of our code is currently limited by the Yee grid to second order; the time integration of the code is also second-order accurate, which is limited by the operator splitting of the fluid, the Yee grid as well as the leapfrog integration of the particles.

## Appendix C. The $R_{31}$ closure and adiabatic coefficients

While the $R_{32}$ closure assumes a fixed adiabatic index $\Gamma$ of 3, the $R_{31}$ closure introduces a term proportional to $\hat{w}$ which alters the pressure equation in such a way that it increases the effective adiabatic index. To show this, we simplify (2.46) by introducing the numerical coefficients $a_w$ and $a_T$ which are defined by comparing

$$\hat{Q} = a_w p_0 \hat{w} + \mathrm{i}\,\text{sign}(k) a_T \hat{T} \tag{C1}$$

with (2.46). Using this ansatz and perturbing the pressure equation (2.32) with $p = p_0 + p_1$, where $p_1$ is the perturbation to the mean pressure $p_0$, in the absence of direct Landau damping ($a_T = 0$), we have

$$\frac{\partial p_1}{\partial t} = (-\Gamma p - a_w p_0)\nabla \cdot w - w \cdot \nabla p = (-\Gamma_{\text{eff}} p_0 - \Gamma p_1)\nabla \cdot w - w \cdot \nabla p, \tag{C2}$$

where $\Gamma_{\text{eff}} = a_w + \Gamma = 4/(4 - \pi) \simeq 4.66$ can be interpreted as the effective adiabatic index of the fluid. The evolution of sound waves of a non-electromagnetic fluid in the
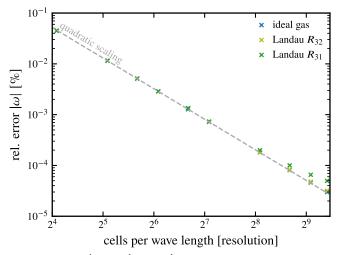
FIGURE 12. Relative error $|(\omega^{\text{sim}} - \omega^{\text{theor}})/\omega^{\text{theor}}|$ of the simulated frequency of a Langmuir wave at $k = 0.05k_{\text{D}}$. The same simulation set-up is used in figure 6, where we use a resolution of 68 cells per wavelength. The resolution here is varied between $68/4 = 17$ and $68 \times 10$ cells per wavelength. The grey line is a reference line for the second-order scaling of the error.

linear regime is governed by the linear term $\Gamma_{\text{eff}} p_0 \nabla \cdot w$ while the term $\Gamma p_1 \nabla \cdot w$ adds nonlinearity to this equation. In the linear approximation, the speed of sound becomes $c_s = (\Gamma_{\text{eff}} p_0 / n_0)^{1/2}$, which coincides with the typical expression for the sound speed $c_s = (\Gamma p_0 / n_0)^{1/2}$ in the limit of $a_w = 0$. This implies that the speed of sound is increased for the $R_{31}$ closure even if direct Landau damping is not present ($a_T = 0$). Interestingly, the effective adiabatic index and the speed of sound are independent of the choice of $\Gamma$. If direct Landau damping, as described by the $R_{31}$ closure, is affecting the fluid (i.e. $a_T \neq 0$), then the effective adiabatic index attains somewhat smaller values in comparison with $a_w + \Gamma$ while the wave frequency becomes complex because of the associated damping. Both are still independent of the choice of $\Gamma$.

This has consequences for simulations that model mildly relativistic fluids. If a simulation set-up includes a fluid with an associated speed of sound near the speed of light $c_s \lesssim c$, then a simulation that uses this set-up with the $R_{31}$ closure can become unstable because $c_s$ can now exceed the speed of light because of the aforementioned reason.

REFERENCES

ALLMANN-RAHN, F., LAUTENBACH, S. & GRAUER, R. 2022 An energy conserving Vlasov solver that tolerates coarse velocity space resolutions: simulation of MMS reconnection events. *J. Geophys. Res.: Space Phys.* **127** (2), e2021JA029976.

ALLMANN-RAHN, F., TROST, T. & GRAUER, R. 2018 Temperature gradient driven heat flux closure in fluid simulations of collisionless reconnection. *J. Plasma Phys.* **84** (3).

BAI, X.-N., CAPRIOLI, D., SIRONI, L. & SPITKOVSKY, A. 2015 Magnetohydrodynamic-particle-in-cell method for coupling cosmic rays with a thermal plasma: application to non-relativistic shocks. *Astrophys. J.* **809** (1), 55.

BAILEY, D.H. 1990 FFTs in external or hierarchical memory. *J. Supercomput.* **4** (1), 23–35.

BELL, A.R. 2004 Turbulent amplification of magnetic field and diffusive shock acceleration of cosmic rays. *Mon. Not. R. Astron. Soc.* **353** (2), 550–558.

BIRDSALL, C.K. & LANGDON, A.B. 1991 *Plasma Physics via Computer Simulation*. Adam Hilger; IOP.

BLANDFORD, R. & EICHLER, D. 1987 Particle acceleration at astrophysical shocks: a theory of cosmic ray origin. *Phys. Rep.* **154** (1), 1–75.

BORIS, J.P., *et al.* 1970 Relativistic plasma simulation-optimization of a hybrid code. In *Proceedings: Fourth Conference on Numerical Simulation of Plasmas*, pp. 3–67. Naval Research Laboratory.

BOULARES, A. & COX, D.P. 1990 Galactic hydrostatic equilibrium with magnetic tension and cosmic-ray diffusion. *Astrophys. J.* **365**, 544.

BOYD, T.J.M. & SANDERSON, J.J. 2003 *The Physics of Plasmas*. Cambridge University Press.

BRAGINSKII, S.I. 1965 Transport processes in a plasma. *Rev. Plasma Phys.* **1**, 205.

BRET, A. & DIECKMANN, M.E. 2010 How large can the electron to proton mass ratio be in particle-in-cell simulations of unstable systems? *Phys. Plasmas* **17** (3), 032109.

BRET, A., GREMILLET, L. & DIECKMANN, M.E. 2010 Multidimensional electron beam-plasma instabilities in the relativistic regime. *Phys. Plasmas* **17** (12), 120501.

BUCK, T., PFROMMER, C., PAKMOR, R., GRAND, R.J.J. & SPRINGEL, V. 2020 The effects of cosmic rays on the formation of milky way-mass galaxies in a cosmological context. *Mon. Not. R. Astron. Soc.* **497** (2), 1712–1737.

BURROWS, R.H., AO, X. & ZANK, G.P. 2014 A new hybrid method. In *Outstanding Problems in Heliophysics: From Coronal Heating to the Edge of the Heliosphere* (ed. Q. Hu & G.P. Zank), Astronomical Society of the Pacific Conference Series, vol. 484, p. 8. Astronomical Society of the Pacific.

BUTCHER, J.C. 2016 *Numerical Methods for Ordinary Differential Equations*, 3rd edn. Wiley.

CAPDEVILLE, G. 2008 A central WENO scheme for solving hyperbolic conservation laws on non-uniform meshes. *J. Comput. Phys.* **227** (5), 2977–3014.

CAPRIOLI, D. & SPITKOVSKY, A. 2014*a* Simulations of ion acceleration at non-relativistic shocks. II. Magnetic field amplification. *Astrophys. J.* **794**, 46.

CAPRIOLI, D. & SPITKOVSKY, A. 2014*b* Simulations of ion acceleration at non-relativistic shocks. III. Particle diffusion. *Astrophys. J.* **794**, 47.

CHEN, G.-Q. & LIU, T.-P. 1993 Zero relaxation and dissipation limits for hyperbolic conservation laws. *Commun. Pure Appl. Maths* **46** (5), 755–781.

COOLEY, J.W. & TUKEY, J.W. 1965 An algorithm for the machine calculation of complex Fourier series. *Math. Comput.* **19** (90), 297–301.

CRAVERO, I., PUPPO, G., SEMPLICE, M. & VISCONTI, G. 2018*a* Cool WENO schemes. *Comput. Fluids* **169**, 71–86.

CRAVERO, I., PUPPO, G., SEMPLICE, M. & VISCONTI, G. 2018*b* CWENO: uniformly accurate reconstructions for balance laws. *Math. Comput.* **87** (312), 1689–1719.

DALGARNO, A. 2006 Interstellar chemistry special feature: the galactic cosmic ray ionization rate. *Proc. Natl Acad. Sci.* **103**, 12269–12273.

DAUGHTON, W., ROYTERSHTEYN, V., KARIMABADI, H., YIN, L., ALBRIGHT, B.J., BERGEN, B. & BOWERS, K.J. 2011 Role of electron physics in the development of turbulent magnetic reconnection in collisionless plasmas. *Nat. Phys.* **7** (7), 539–542.

DAUGHTON, W., SCUDDER, J. & KARIMABADI, H. 2006 Fully kinetic simulations of undriven magnetic reconnection with open boundary conditions. *Phys. Plasmas* **13** (7), 072101.

DAWSON, J. 1962 One-dimensional plasma model. *Phys. Fluids* **5** (4), 445.

DELLACHERIE, S. 2010 Analysis of Godunov type schemes applied to the compressible Euler system at low Mach number. *J. Comput. Phys.* **229** (4), 978–1016.

DIMITS, A.M., JOSEPH, I. & UMANSKY, M.V. 2014 A fast non-Fourier method for Landau-fluid operators. *Phys. Plasmas* **21** (5), 055907.

DING, H., LI, C. & CHEN, Y.Q. 2015 High-order algorithms for Riesz derivative and their applications (II). *J. Comput. Phys.* **293**, 218–237.

DRAINE, B.T. 2011 *Physics of the Interstellar and Intergalactic Medium*. Princeton University Press.

EINFELDT, B., MUNZ, C.D., ROE, P.L. & SJÖGREEN, B. 1991 On Godunov-type methods near low densities. *J. Comput. Phys.* **92** (2), 273–295.

FARBER, R., RUSZKOWSKI, M., YANG, H.-Y.K. & ZWEIBEL, E.G. 2018 Impact of cosmic ray transport on galactic winds. *Astrophys. J.* **856** (2), 112.

FINELLI, F., CERRI, S.S., CALIFANO, F., PUCCI, F., LAVEDER, D., LAPENTA, G. & PASSOT, T. 2021 Bridging hybrid- and full-kinetic models with Landau-fluid electrons. I. 2D magnetic reconnection. *Astron. Astrophys.* **653**, A156.

GARGATÉ, L., BINGHAM, R., FONSECA, R.A. & SILVA, L.O. 2007 dHybrid: a massively parallel code for hybrid simulations of space plasmas. *Comput. Phys. Commun.* **176** (6), 419–425.

GIRICHIDIS, P., NAAB, T., HANASZ, M. & WALCH, S. 2018 Cooler and smoother - the impact of cosmic rays on the phase structure of galactic outflows. *Mon. Not. R. Astron. Soc.* **479**, 3042–3067.

GUO, F. & OH, S.P. 2008 Feedback heating by cosmic rays in clusters of galaxies. *Mon. Not. R. Astron. Soc.* **384** (1), 251–266.

HAKIM, A., LOVERICH, J. & SHUMLAK, U. 2006 A high resolution wave propagation scheme for ideal two-fluid plasma equations. *J. Comput. Phys.* **219** (1), 418–442.

HAMMETT, G.W. & PERKINS, F.W. 1990 Fluid moment models for Landau damping with application to the ion-temperature-gradient instability. *Phys. Rev. Lett.* **64** (25), 3019–3022.

HANASZ, M., LESCH, H., NAAB, T., GAWRYSZCZAK, A., KOWALIK, K. & WÓLTAŃSKI, D. 2013 Cosmic rays can drive strong outflows from gas-rich high-redshift disk galaxies. *Astrophys. J. Lett.* **777** (2), L38.

HARTEN, A. & HYMAN, J.M. 1983 Self adjusting grid methods for one-dimensional hyperbolic conservation laws. *J. Comput. Phys.* **50** (2), 235–269.

HOCKNEY, R.W. 1988 *Computer Simulation Using Particles*. CRC Press.

HOLCOMB, C.J. 2019 The microphysics of gyroresonant streaming instabilities and cosmic ray self-confinement. PhD thesis, Princeton University.

HOLCOMB, C. & SPITKOVSKY, A. 2019 On the growth and saturation of the gyroresonant streaming instabilities. *Astrophys. J.* **882** (1), 3.

HOLLWEG, J.V. 1971 Nonlinear Landau damping of Alfvén waves. *Phys. Rev. Lett.* **27** (20), 1349–1352.

HONG, J., LEE, E., MIN, K. & PARKS, G.K. 2012 Effect of ion-to-electron mass ratio on the evolution of ion beam driven instability in particle-in-cell simulations. *Phys. Plasmas* **19** (9), 092111.

HSIAO, L. 1997 *Quasilinear Hyperbolic Systems and Dissipative Mechanisms*. World Scientific.

HUNANA, P., TENERANI, A., ZANK, G.P., GOLDSTEIN, M.L., WEBB, G.M., KHOMENKO, E., COLLADOS, M., CALLY, P.S., ADHIKARI, L. & VELLI, M. 2019*a* An introductory guide to fluid models with anisotropic temperatures. Part 2. Kinetic theory, Padé approximants and Landau fluid closures. *J. Plasma Phys.* **85** (6).

HUNANA, P., TENERANI, A., ZANK, G.P., KHOMENKO, E., GOLDSTEIN, M.L., WEBB, G.M., CALLY, P.S., COLLADOS, M., VELLI, M. & ADHIKARI, L. 2019*b* An introductory guide to fluid models with anisotropic temperatures. Part 1. CGL description and collisionless fluid hierarchy. *J. Plasma Phys.* **85** (6), 205850602.

HUNANA, P., ZANK, G.P., LAURENZA, M., TENERANI, A., WEBB, G.M., GOLDSTEIN, M.L., VELLI, M. & ADHIKARI, L. 2018 New closures for more precise modeling of Landau damping in the fluid framework. *Phys. Rev. Lett.* **121** (13), 135101.

JACOB, S. & PFROMMER, C. 2017 Cosmic ray heating in cool core clusters – I. Diversity of steady state solutions. *Mon. Not. R. Astron. Soc.* **467** (2), 1449–1477.

JI, S., CHAN, T.K., HUMMELS, C.B., HOPKINS, P.F., STERN, J., KEREŠ, D., QUATAERT, E., FAUCHER-GIGUÈRE, C.-A. & MURRAY, N. 2020 Properties of the circumgalactic medium in cosmic ray-dominated galaxy haloes. *Mon. Not. R. Astron. Soc.* **496** (4), 4221–4238.

JIANG, G.-S. & SHU, C.-W. 1996 Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* **126** (1), 202–228.

KULSRUD, R. & PEARCE, W.P. 1969 The effect of wave-particle interactions on the propagation of cosmic rays. *Astrophys. J.* **156**, 445.

LANGDON, A.B. & BIRDSALL, C.K. 1970 Theory of plasma simulation using finite-size particles. *Phys. Fluids* **13** (8), 2115–2122.

LEE, M.A. & VÖLK, H.J. 1973 Damping and nonlinear wave-particle interactions of Alfvén-waves in the solar wind. *Astrophys. Space Sci.* **24** (1), 31–49.

LEVEQUE, R.J. 2002 *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press.

LEVEQUE, R.J. & PELANTI, M. 2001 A class of approximate riemann solvers and their relation to relaxation schemes. *J. Comput. Phys.* **172** (2), 572–591.

LIPATOV, A.S. 2002 *The Hybrid Multiscale Simulation Technology*. Springer.

MARCOWITH, A., BRET, A., BYKOV, A., DIECKMAN, M.E., O'C DRURY, L., LEMBÈGE, B., LEMOINE, M., MORLINO, G., MURPHY, G., PELLETIER, G., PLOTNIKOV, I., REVILLE, B., RIQUELME, M., SIRONI, L. & STOCKEM NOVO, A. 2016 The microphysics of collisionless shock waves. *Rep. Prog. Phys.* **79** (4), 046901.

MARCOWITH, A., VAN MARLE, A.J. & PLOTNIKOV, I. 2021 The cosmic ray-driven streaming instability in astrophysical and space plasmas. *Phys. Plasmas* **28** (8), 080601.

VAN MARLE, A.J., CASSE, F. & MARCOWITH, A. 2018 On magnetic field amplification and particle acceleration near non-relativistic astrophysical shocks: particles in MHD cells simulations. *Mon. Not. R. Astron. Soc.* **473** (3), 3394–3409.

MORENO, Q., DIECKMANN, M.E., RIBEYRE, X., JEQUIER, S., TIKHONCHUK, V.T. & D'HUMIÈRES, E. 2018 Impact of the electron to ion mass ratio on unstable systems in particle-in-cell simulations. *Phys. Plasmas* **25** (6), 062125.

NG, J., HAKIM, A., WANG, L. & BHATTACHARJEE, A. 2020 An improved ten-moment closure for reconnection and instabilities. *Phys. Plasmas* **27** (8), 082106.

PADOVANI, M., IVLEV, A.V., GALLI, D., OFFNER, S.S.R., INDRIOLO, N., RODGERS-LEE, D., MARCOWITH, A., GIRICHIDIS, P., BYKOV, A.M. & KRUIJSSEN, J.M.D. 2020 Impact of low-energy cosmic rays on star formation. *Space Sci. Rev.* **216**, 29.

PAKMOR, R., PFROMMER, C., SIMPSON, C.M. & SPRINGEL, V. 2016 Galactic winds driven by isotropic and anisotropic cosmic-ray diffusion in disk galaxies. *Astrophys. J.* **824**, L30.

PARK, W., PARKER, S., BIGLARI, H., CHANCE, M., CHEN, L., CHENG, C.Z., HAHM, T.S., LEE, W.W., KULSRUD, R., MONTICELLO, D., SUGIYAMA, L. & WHITE, R. 1992 Three-dimensional hybrid gyrokinetic-magnetohydrodynamics simulation. *Phys. Fluids* B **4** (7), 2033–2037.

PASSOT, T., HENRI, P., LAVEDER, D. & SULEM, P.-L. 2014 Fluid simulations of ion scale plasmas with weakly distorted magnetic fields. *Eur. Phys. J.* D **68** (7), 207.

PEERY, K. & IMLAY, S. 1988 Blunt-body flow simulations. In *24th Joint Propulsion Conference*. American Institute of Aeronautics and Astronautics.

PFROMMER, C. 2013 Toward a comprehensive model for feedback by active galactic nuclei: new insights from M87 observations by LOFAR, Fermi, and H.E.S.S. *Astrophys. J. Lett.* **779** (1), 10.

PFROMMER, C., PAKMOR, R., SCHAAL, K., SIMPSON, C.M. & SPRINGEL, V. 2017 Simulating cosmic ray physics on a moving mesh. *Mon. Not. R. Astron. Soc.* **465** (4), 4500–4529.

RIQUELME, M.A. & SPITKOVSKY, A. 2009 Nonlinear study of Bell's cosmic ray current-driven instability. *Astrophys. J.* **694**, 626–642.

ROE, P.L. 1981 Approximate Riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.* **43** (2), 357–372.

RUSZKOWSKI, M., YANG, H.-Y.K. & REYNOLDS, C.S. 2017 Cosmic-ray feedback heating of the intracluster medium. *Astrophys. J.* **844** (1), 13.

RUSZKOWSKI, M., YANG, H.Y.K. & ZWEIBEL, E. 2017 Global simulations of galactic winds including cosmic-ray streaming. *Astrophys. J. Lett.* **834** (2), 208.

SHALABY, M., BRODERICK, A.E., CHANG, P., PFROMMER, C., LAMBERTS, A. & PUCHWEIN, E. 2017*a* Importance of resolving the spectral support of beam-plasma instabilities in simulations. *Astrophys. J.* **848** (2), 81.

SHALABY, M., BRODERICK, A.E., CHANG, P., PFROMMER, C., LAMBERTS, A. & PUCHWEIN, E. 2017*b* SHARP: a spatially higher-order, relativistic particle-in-cell code. *Astrophys. J.* **841** (1), 52.

SHALABY, M., BRODERICK, A.E., CHANG, P., PFROMMER, C., LAMBERTS, A. & PUCHWEIN, E. 2018 Growth of beam-plasma instabilities in the presence of background inhomogeneity. *Astrophys. J.* **859** (1), 45.

SHALABY, M., BRODERICK, A.E., CHANG, P., PFROMMER, C., PUCHWEIN, E. & LAMBERTS, A. 2020 The growth of the longitudinal beam–plasma instability in the presence of an inhomogeneous background. *J. Plasma Phys.* **86** (2).

SHALABY, M., LEMMERZ, R., THOMAS, T. & PFROMMER, C. 2022 The mechanism of efficient electron acceleration at parallel nonrelativistic shocks. *Astrophys. J.* **932** (2), 86.

SHALABY, M., THOMAS, T. & PFROMMER, C. 2021 A new cosmic-ray-driven instability. *Astrophys. J.* **908** (2), 206.

SHALABY, M., THOMAS, T., PFROMMER, C., LEMMERZ, R. & BRESCI, V. 2023 Deciphering the physical basis of the intermediate-scale instability. e-prints, arXiv:2305.18050.

SHUMLAK, U., LILLY, R., REDDELL, N., SOUSA, E. & SRINIVASAN, B. 2011 Advanced physics calculations using a multi-fluid plasma model. *Comput. Phys. Commun.* **182** (9), 1767–1770.

SHUMLAK, U. & LOVERICH, J. 2003 Approximate Riemann solver for the two-fluid plasma model. *J. Comput. Phys.* **187** (2), 620–638.

SIMON, S. & MANDAL, J.C. 2019 A simple cure for numerical shock instability in the HLLC Riemann solver. *J. Comput. Phys.* **378**, 477–496.

SIMPSON, C. M., PAKMOR, R., MARINACCI, F., PFROMMER, C., SPRINGEL, V., GLOVER, S.C.O., CLARK, P.C. & SMITH, R.J. 2016 The role of cosmic-ray pressure in accelerating galactic outflows. *Astrophys. J.* **827**, L29.

SIRONI, L. & SPITKOVSKY, A. 2014 Relativistic reconnection: an efficient source of non-thermal particles. *Astrophys. J.* **783** (1), L21.

SNYDER, P.B., HAMMETT, G.W. & DORLAND, W. 1997 Landau fluid models of collisionless magnetohydrodynamics. *Phys. Plasmas* **4** (11), 3974–3985.

SOARES FRAZAO, S. & ZECH, Y. 2002 Undular bores and secondary waves -experiments and hybrid finite-volume modelling. *J. Hydraul. Res.* **40** (1), 33–43.

SPITKOVSKY, A. 2008 Particle acceleration in relativistic collisionless shocks: fermi process at last? *Astrophys. J.* **682** (1), L5.

STIX, T.H. 1992 *Waves in Plasmas*. AIP.

STRANG, G. 1968 On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.* **5** (3), 506–517.

TAKAHASHI, D. & KANADA, Y. 2000 High-performance radix-2, 3 and 5 parallel 1-D complex FFT algorithms for distributed-memory parallel computers. *J. Supercomput.* **15** (2), 207–228.

THOMAS, T. & PFROMMER, C. 2019 Cosmic-ray hydrodynamics: Alfvén-wave regulated transport of cosmic rays. *Mon. Not. R. Astron. Soc.* **485** (3), 2977–3008.

THOMAS, T., PFROMMER, C. & ENßLIN, T.A. 2020 Probing cosmic ray transport with radio synchrotron harps in the Galactic center. *Astrophys. J.* **890** (2), L18.

THOMAS, T., PFROMMER, C. & PAKMOR, R. 2022 Cosmic ray-driven galactic winds: transport modes of cosmic rays and Alfvén-wave dark regions. *Mon. Not. R. Astron. Soc.* **521** (2), 3023–3042.

TORO, E.F. 2009 *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*, 3rd edn. Springer.

TORO, E.F., SPRUCE, M. & SPEARES, W. 1994 Restoration of the contact surface in the HLL-Riemann solver. *Shock Waves* **4** (1), 25–34.

UHLIG, M., PFROMMER, C., SHARMA, M., NATH, B.B., ENßLIN, T.A. & SPRINGEL, V. 2012 Galactic winds driven by cosmic ray streaming. *Mon. Not. R. Astron. Soc.* **423**, 2374–2396.

UMANSKY, M.V., DIMITS, A.M., JOSEPH, I., OMOTANI, J.T. & ROGNLIEN, T.D. 2015 Modeling of tokamak divertor plasma for weakly collisional parallel electron transport. *J. Nucl. Mater.* **463**, 506–509.

WANG, L., HAKIM, A.H., BHATTACHARJEE, A. & GERMASCHEWSKI, K. 2015 Comparison of multi-fluid moment models with particle-in-cell simulations of collisionless magnetic reconnection. *Phys. Plasmas* **22** (1), 012108.

WANG, L., HAKIM, A.H., NG, J., DONG, C. & GERMASCHEWSKI, K. 2020 Exact and locally implicit source term solvers for multifluid-Maxwell systems. *J. Comput. Phys.* **415**, 109510.

WANG, L., ZHU, B., XU, X.-Q. & LI, B. 2019 A Landau-fluid closure for arbitrary frequency response. *AIP Adv.* **9** (1), 015217.

WHITHAM, G.B. 1974 *Linear and Nonlinear Waves*. Wiley.

XIE, H.-S. 2014 PDRF: a general dispersion relation solver for magnetized multi-fluid plasma. *Comput. Phys. Commun.* **185** (2), 670–675.

ZWEIBEL, E.G. 2017 The basis for cosmic ray feedback: written on the wind. *Phys. Plasmas* **24** (5), 055402.