Comparisons of time explicit hybrid kinetic-fluid code Architect for Plasma Wakefield Acceleration with a full PIC code

F. Massimo[a,b,*], S. Atzeni[b] and A. Marocchino[b,c]

[a] *Laboratoire d'Optique Appliquée, ENSTA ParisTech, CNRS, École Polytechnique, Université Paris-Saclay, 828 bd des Maréchaux, 91762 Palaiseau - France.*

[b] *Dipartimento SBAI, Università di Roma "La Sapienza", Via A. Scarpa 14, 00161 Roma, Italy*

[c] *INFN - LNF, via Enrico Fermi 40, 00044 Frascati, Italy*

Abstract

Architect, a time explicit hybrid code designed to perform quick simulations for electron driven plasma wakefield acceleration, is described. In order to obtain beam quality acceptable for applications, control of the beam-plasma-dynamics is necessary. Particle in Cell (PIC) codes represent the state-of-the-art technique to investigate the underlying physics and possible experimental scenarios; however PIC codes demand the necessity of heavy computational resources. Architect code substantially reduces the need for computational resources by using a hybrid approach: relativistic electron bunches are treated kinetically as in a PIC code and the background plasma as a fluid. Cylindrical symmetry is assumed for the solution of the electromagnetic fields and fluid equations. In this paper both the underlying algorithms as well as a comparison with a fully three dimensional particle in cell code are reported. The comparison highlights the good agreement between the two models up to the weakly non linear regimes. In highly non linear regimes the two models only disagree in a localized region, where the plasma electrons expelled by the bunch close up at the end of the first plasma oscillation.

Keywords: Particle in Cell codes, Plasma Wakefield Acceleration, hybrid codes

1. Introduction

Plasma acceleration [1] is a particle acceleration technology which could overcome the limits of conventional radio-frequency accelerators, where the accelerating gradients are limited by the breakdown threshold of metallic cavities. In particular the Plasma Wakefield Acceleration (PWFA) scheme demonstrated high energy gains through the excitation of wakefields by high energy electron bunches injected in a plasma [2]. In this scheme a bunch - called driver - injected in a plasma channel excites a wakefield that can be used to accelerate a second bunch - called witness - injected in the proper wakefield phase. The use of plasma-accelerated electron beams for applications is presently limited by high energy spread

---

and emittance. Considerable interest is thus shown in the improvement of electron PWFA performances [3], aiming to reliably obtain beam quality comparable to those of conventional acceleration techniques.

The state-of-art numerical method to study PWFA are Particle in Cell (PIC) codes [4], which self-consistently integrate the Vlasov-Maxwell system of equations. However the investigation of the beam-plasma interactions with PIC codes often requires time-consuming simulations, even with efficient parallelization techniques.

An alternative hybrid kinetic-fluid technique which considerably reduces the simulation time consists in modeling the background plasma as a fluid, as successfully demonstrated by the codes INF&RNO [5] for Laser Wake Field Acceleration (LWFA) acceleration and in LCODE [6] for PWFA.

Considerable progress in the efficiency of simulations for plasma acceleration can be obtained with the quasi-static approximation (QSA) [7, 8, 9], which takes advantage of the different evolution timescales of relativistic beams and background plasma. PIC codes successfully demonstrated run time reduction using QSA [7, 10, 11, 12]. QSA has been successfully implemented also in hybrid kinetic-fluid codes, as INF&RNO [5] (which supports both time-explicit and QSA simulation modes) and LCODE [6]. The QSA is based on the assumption that plasma quantities as electron density are predominantly dependent on the comoving variable $\xi = z - ct$ (where z is the beam propagation direction, t the time and c the speed of light), thus in PWFA self-consistent initialization with beams injected from outside the plasma cannot be accurately studied, especially in presence of sharp plasma density gradients.

To efficiently study PWFA up to the weakly nonlinear regimes [13, 14], the hybrid kinetic-fluid code Architect has been developed, as briefly described in [15, 16, 17]. The time-explicit character of Architect allows to self-consistently simulate physical set-ups in which QSA assumptions are not satisfied as in the case of beam injection in a plasma. The electromagnetic fields of the beam entering the plasma from vacuum are initialized solving the Poisson problem in the beam rest frame, providing the correct initial conditions for the routines which solve the electromagnetic field and fluid equations. The hybrid model of Architect treats the relativistic electron beams with standard PIC techniques and the background plasma electrons as a fluid. The ions are modeled as a static background. The simulation time is considerably reduced with respect to a fully kinetic code and no parallelization is necessary. The present implementation of Architect integrates the model equations in 2D assuming a cylindrical symmetry, as typical PWFA experimental set-ups have cylindrical symmetry. However, the model can be easily extended up to three dimensions. In this paper we present comparisons of Architect results in the weakly nonlinear regime of PWFA against full 3D PIC simulations with the code ALaDyn [14, 18, 19, 20]. Although comparisons between a hybrid kinetic-fluid code have been shown with the code INF&RNO for Laser Wakefield Acceleration in [5], to our knowledge no similar systematic comparisons have been shown for electron PWFA. Both INF&RNO and Architect solve the same plasma background equations: the relativistic cold fluid equations in a time-explicit formulation. A part from the different driver nature, i.e. a laser pulse in INF&RNO and an electron beam in Architect, the fluid and field solvers of INF&RNO use non-staggered quantities and Runge-Kutta integrators while the ones in Architect are based on finite difference time explicit domain schemes.

Architect has been developed in the framework of the PWFA experiments planned at the

SPARC_LAB facility in Frascati - Italy [21, 22].

The paper is organised as follows. In the second section we introduce the code hybrid model. In the third section we summarize the code loop for each integration timestep, including the schemes used to integrate the model equations. In the fourth section we show a comparison of the results of Architect and the 3D full PIC code ALaDyn.

2. The hybrid: fluid-kinetic model

Architect solves the plasma wakefield acceleration system evolution by combining a particle-in-cell (PIC) approach [4] with fluid equations. The PIC-like approach, the most computationally demanding, is used only to evolve the electron bunch(es); particles are evolved in the 3D3V phase-space. The background plasma evolves accordingly to cold relativistic fluid equations in the assumption of cylindrical $r-z$ symmetry, with axes schematic depicted in Fig.1. The induced bunch current as well as the background plasma current are used as source terms of the Maxwell's equations, also solved in cylindrical symmetry.

Architect uses dimensionless units: time is in unit of the inverse plasma frequency ω_p^{-1}; velocities are normalised to the speed of light c, lengths are in units of k_p^{-1} ($k_p = \omega_p/c$). Momenta are taken in units of $m_e c$, with m_e the electron mass. As a consequence, density is in units of n_0, the initial nominal background density. Electromagnetic fields are normalised as $\mathbf{E} = \widetilde{\mathbf{E}}E_0$ and $\mathbf{B} = \widetilde{\mathbf{B}}E_0/c$, with $E_0 = m_e\omega_p c/e$ the cold linear wavebreaking limit and e the electron charge. Bunch particles are evolved in a 3D3V space:

$$
\begin{aligned}
\frac{d\widetilde{\mathbf{x}}_{\mathrm{p}}}{d\widetilde{t}} &= \boldsymbol{\beta}_{\mathrm{p}}, \\
\frac{d\widetilde{p}_{\mathrm{p},x}}{d\widetilde{t}} &= -\left(\widetilde{E}_{\mathrm{r}} - \beta_{\mathrm{p,z}}\widetilde{B}_{\varphi}\right)\cos\theta, \\
\frac{d\widetilde{p}_{\mathrm{p},y}}{d\widetilde{t}} &= -\left(\widetilde{E}_{\mathrm{r}} - \beta_{\mathrm{p,z}}\widetilde{B}_{\varphi}\right)\sin\theta, \\
\frac{d\widetilde{p}_{\mathrm{p},z}}{d\widetilde{t}} &= -\widetilde{E}_{\mathrm{z}} - \widetilde{B}_{\varphi}\left(\beta_{\mathrm{p,x}}\cos\theta + \beta_{\mathrm{pi,y}}\sin\theta\right),
\end{aligned}
\tag{1}
$$

where dimensionless quantities are denoted with a tilda-notation, while the subscript p indicates that each set of equations is applied to the p-th particle, for all the particles. Equation (1) already takes into account the cylindrical symmetry used to discretised the electromagnetic fields. The trigonometric projections are $\cos\theta = \widetilde{x}_{\mathrm{p}}/\sqrt{\widetilde{x}_{\mathrm{p}}^2 + \widetilde{y}_{\mathrm{p}}^2}$ and $\sin\theta = \widetilde{y}_{\mathrm{p}}/\sqrt{\widetilde{x}_{\mathrm{p}}^2 + \widetilde{y}_{\mathrm{p}}^2}$, respectively. The use of full six dimensional phase space maintains a standard definition for beam kinetic parameters such as transverse-x-y emittance (ε_x, ε_y) and transverse-x-y rms-size (σ_x, σ_y).

The background plasma electrons are instead evolved with a relativistic cold fluid approximation,

$$
\begin{aligned}
\frac{\partial\widetilde{n}_e}{\partial\widetilde{t}} + \widetilde{\nabla}\cdot(\widetilde{n}_e\boldsymbol{\beta}_e) &= 0, \\
\frac{\partial\widetilde{p}_{\mathrm{e,r}}}{\partial\widetilde{t}} + \widetilde{\nabla}\cdot(\widetilde{p}_{\mathrm{e,r}}\boldsymbol{\beta}_e) &= -\widetilde{E}_{\mathrm{r}} + \beta_{\mathrm{e,z}}\widetilde{B}_{\varphi}, \\
\frac{\partial\widetilde{p}_{\mathrm{e,z}}}{\partial\widetilde{t}} + \widetilde{\nabla}\cdot(\widetilde{p}_{\mathrm{e,z}}\boldsymbol{\beta}_e) &= -\widetilde{E}_{\mathrm{z}} - \beta_{\mathrm{e,r}}\widetilde{B}_{\varphi};
\end{aligned}
\tag{2}
$$

3

where the first equation is the continuity equation, the second and third represent the r and z components of the momentum equation; $\boldsymbol{\beta}_e$ is the relativistic β for the background plasma. The background plasma ions are instead treated as an immobile background.

The electromagnetic fields are evolved accordingly to the Maxwell's equations, with the current density given by the sum of the background current density and the bunch current density:

$$
\begin{aligned}
\frac{\partial \widetilde{B}_\varphi}{\partial \widetilde{t}} &= \partial_{\widetilde{r}} \widetilde{E}_z - \partial_{\widetilde{z}} \widetilde{E}_r, \\
\frac{\partial \widetilde{E}_r}{\partial \widetilde{t}} &= -\frac{\partial \widetilde{B}_\varphi}{\partial \widetilde{z}} - \widetilde{n}_e \beta_{e,r} - \widetilde{n}_b \beta_{b,r}, \\
\frac{\partial \widetilde{E}_z}{\partial \widetilde{t}} &= \frac{1}{\widetilde{r}} \frac{\partial (\widetilde{r} \widetilde{B}_\varphi)}{\partial \widetilde{r}} - \widetilde{n}_e \beta_{e,z} - \widetilde{n}_b \beta_{b,z},
\end{aligned} \tag{3}
$$

where $\boldsymbol{\beta}_b$ is the bunch relativistic β and $\widetilde{n}_b$ is the bunch density.
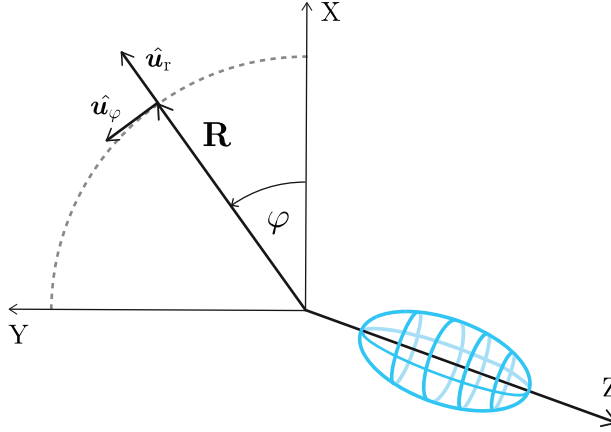

Figure 1: Cylindrical and cartesian axes in Architect. Superimposed is a gaussian bunch, whose particles move in 3D.

Since Maxwell's and hydrodynamic equations are solved in cylindrical symmetry, despite bunches evolve in a full 3D3V phase space, simulated bunches need to exhibit cylindrical symmetry. For bunches with marked transverse ellipticity the code predicitions become inaccurate.

3. General code loop

To describe the code general structure, we can break a simulation timestep into five subsequent steps. The first step consists in deriving the bunch current by projecting bunch particles onto the $r - z$ grid. The second step consists in computing the background current from the fluid number density and momentum. The third step corresponds to a finite difference time domain integration of Maxwell's equations with a Yee scheme, the source terms represented by the current are known from the previous steps. We notice that Architect computes the total electromagnetic fields, the sum of bunch and background induced

fields. The fourth step integrates Eq.(2), to update number density and momentum, by using the just computed electromagnetic fields. The fifth and final step consists in particle time advancement. The loop is depicted and graphically summarised in Fig. 2.
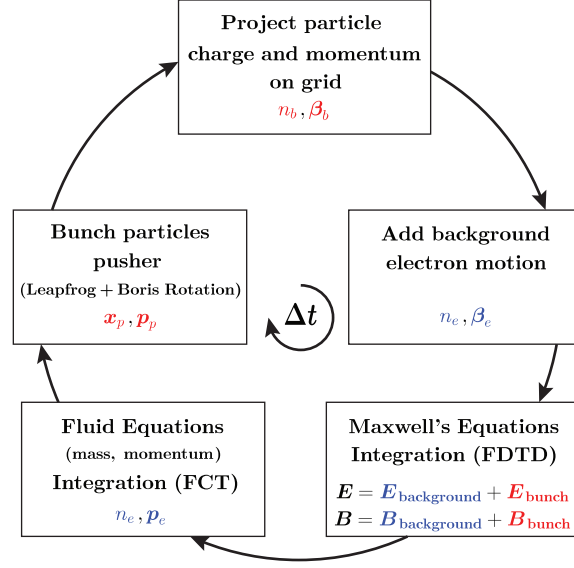

Figure 2: Architect Loop. Quantities referred to the background plasma are highlighted in blue, while quantities referred to the electron beam are highligted in red.

The electron bunches are initialized in vacuum at the entrance of the plasma channel. To properly study the bunch transition from vacuum to plasma, the initial electromagnetic fields of the beam-plasma system must be accurately computed. As discussed in [23], using initial null or incorrect fields leads to an unphysical evolution of the overall system. Since the plasma background electrons and ions are initialized at rest and equally distributed, i.e. $n_e(r, z, t = 0) = n_i(r, z, t = 0)$, where n_i is the ion number density, the only electromagnetic fields to be computed at $t = 0$ are the self consistent bunch EM fields. The computation of these fields is performed through a Poisson solver in the beam rest frame as in [24, 25] and is discussed in detail in [16].

In the time loop of Fig. 2, Eqs.(2)-(3) are solved on a uniform 2D grid (in r and z) as depicted in Fig.3. As shown in Fig. 3, fluid and electromagnetic quantities have different centring: while the Yee-lattice [26] centers electric fields on edges and the magnetic field on faces, the whole fluid quantities are centered as the longitudinal electric field E_z. Cell sizes are chosen depending on the initial bunch rms-dimensions. This choice allows the user to control the number of points for each rms-dimension so to well catch the right evolution spatial scale. The overall time centering of the physical quantities in the code is shown in Fig. 4. The choice of time-centring allows to easily write both the backgound $\widetilde{\mathbf{J}}_e$ as well as the bunch current $\widetilde{\mathbf{J}}_b$:

$$\widetilde{\mathbf{J}}^n_{b_{i,j}} = \widetilde{n}^n_{b_{i,j}}\boldsymbol{\beta}^n_{b_{i,j}}, \ \widetilde{\mathbf{J}}^n_{e_{i,j}} = \widetilde{n}^n_{e_{i,j}}\boldsymbol{\beta}^n_{e_{i,j}}, \tag{4}$$

where $\boldsymbol{\beta}_e$ is the background velocity, $\boldsymbol{\beta}_b$ the bunch velocity.

We recall that bunches are kinetically-PIC discretised, consequently the bunch current needs to be computed by projecting velocity onto the $r - z$ grid [27]. PIC leapfrog scheme
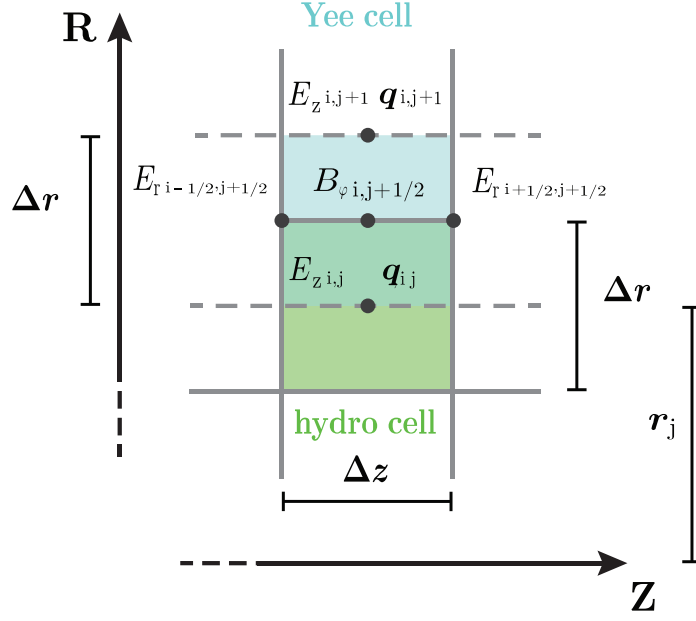
5

Figure 3: Schematic of physical quantity spatial centering, highlighting the indices of the fluid quantities (i.e. density and momentum, denoted with **q**) and the electromagnetic quantities. The electromagnetic quantity are centered accordingly to the Yee lattice: the poloidal magnetic field is cell-centered while currents and electric fields are face-centered. The fluid quantities are instead all centered as the longitudinal electric field E_z.
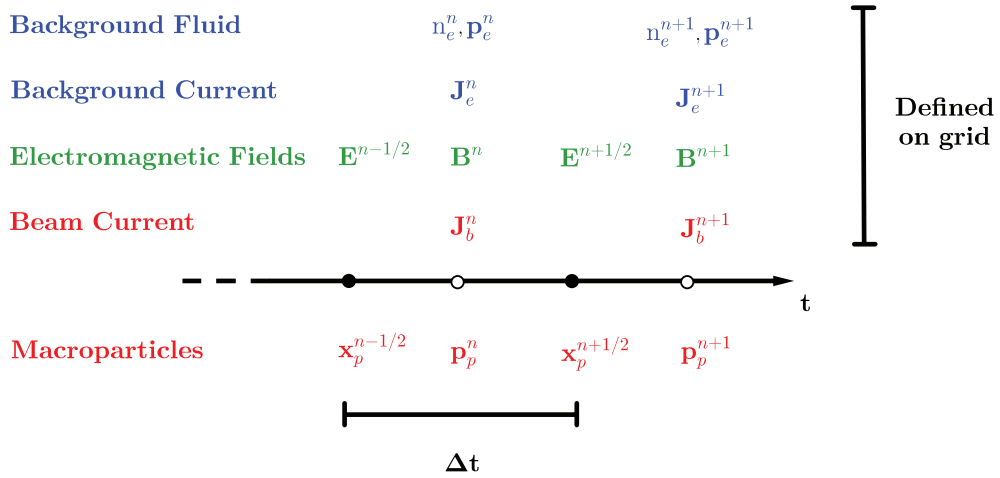

Figure 4: Schematic of physical quantity time centering.

naturally centers velocity at integer timestep, while we compute the particle position by a standard mean approach: $\widetilde{\mathbf{x}}_{\mathrm{p}}^{n} = \left(\widetilde{\mathbf{x}}_{\mathrm{p}}^{n-1/2} + \widetilde{\mathbf{x}}_{\mathrm{p}}^{n+1/2} \right)/2$. Figure 5 sketches the computation of the background and bunch current steps.

Equations (3) are solved with an electromagnetic second order Leap-Frog scheme, the Finite Difference Time Domain (FDTD) [26]. Figure 6 highlights the old and the updated quantities in this step. The electric field $\widetilde{\mathbf{E}}$ is thus shifted by half an integration timestep with respect to the magnetic field $\widetilde{\mathbf{B}}$, the electromagnetic current of the beam particles $\widetilde{\mathbf{J}}_{\mathrm{b}}$ and the electromagnetic current of the background electrons $\widetilde{\mathbf{J}}_{\mathrm{e}}$. Since all the fluid quantities are centered in space as $\widetilde{E}_{\mathrm{z}}$, linear interpolation is used to compute the radial background current at the same position of $\widetilde{E}_{\mathrm{r}}$; this allows to solve the discretized version the second equation of Eqs.(3). To ensure that Gauss's law is satisfied after the electromagnetic advance, Boris correction [4] is used.

Once the forces acting on each of the beam particles are extrapolated from the grid as discussed in [4], Eqs.(1) are then solved through classical second order explicit Leap-Frog scheme with Boris rotation method [4]; in this scheme beam particle positions $\widetilde{\mathbf{x}}_{\mathrm{p}}$ and momenta $\widetilde{\mathbf{p}}_{\mathrm{p}}$ are shifted by half timestep.

Equations (2) are solved with an operator splitting technique. To perform the advection, both an upwind and the shock-capturing scheme Flux Corrected Transport (FCT) [28, 29] are implemented in the code [16].

Figures 5, 6 and 7 summarize the code loop, highlighting the known quantities and the computed quantities at each step. It is assumed that the known quantities at the beginning of each loop iteration are: electric field $\mathbf{E}^{n-1/2}$ and positions $\mathbf{x}_{p}^{n-1/2}$ of the particles at timestep $n-1/2$, magnetic field $\mathbf{B}^{n}$, particles momenta $\mathbf{p}_{p}^{n}$, fluid density n_{e}^{n} and momentum $\mathbf{p}_{e}^{n}$ at timestep n, plus the particles positions at timestep $n+1/2$. The bunch current $\mathbf{J}_{b}^{n}$ is computed from the known positions and the momenta of the particles, while the background plasma current $\mathbf{J}_{e}^{n}$ is computed from the electron fluid density and momentum (Fig. 5). The electromagnetic fields $\mathbf{E}^{n+1/2}$ and $\mathbf{B}^{n}$ can be advanced with the FDTD scheme from the currents computed at the previous step and the known fields (Fig. 6). From the updated fields the force acting on each particle and on each fluid cell can be computed, to update the particle positions and momenta throught the Boris Pusher and the fluid quantities with the upwind or FCT scheme (Fig. 7). The loop can be iterated.

4. Comparison with 3D full PIC code

In this section we compare Architect versus the 3D full PIC code ALaDyn [18, 14, 19, 20] for a series of cases in the weakly nonlinear regime, up to the transition to non linear regimes, defined as follows.

In 1D cases a key parameter to evaluate the departure from linearity is the α-value, where $\alpha = n_{b,\,peak}/n_{0}$ with $n_{b,\,peak}$ the peak bunch density and n_{0} the initial plasma density, since it allows to distinguish among the different regimes [30]: linear, weakly-non-linear and non-linear regime. In three dimensions the same role is played by the normalized charge $\widetilde{Q}$, defined as the the ratio of the number of bunch electrons and the plasma electrons in a cubic skin depth k_{p}^{-3} [31]. Linear regimes, weakly nonlinear regimes and highly nonlinear regimes correspond to normalized charge values $\widetilde{Q} \ll 1$, $\widetilde{Q} \lesssim 1$, $\widetilde{Q} \gg 1$ respectively. For a
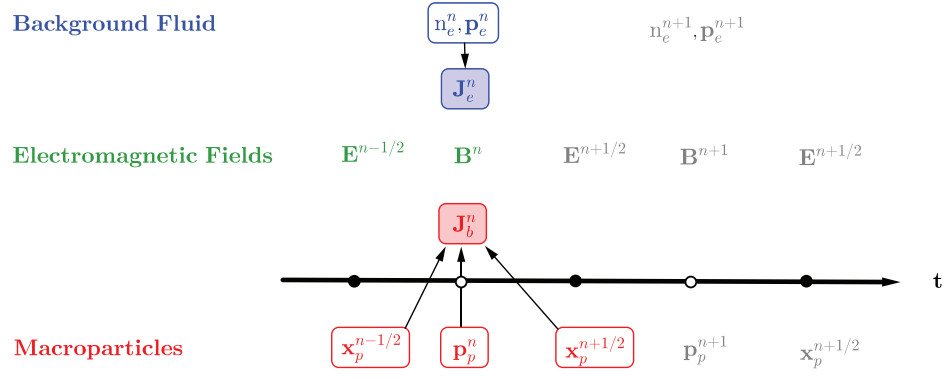
Figure 5: Beam and fluid current computation on the grid, highlighting the known (empty boxes) and the computed quantities (coloured boxes).
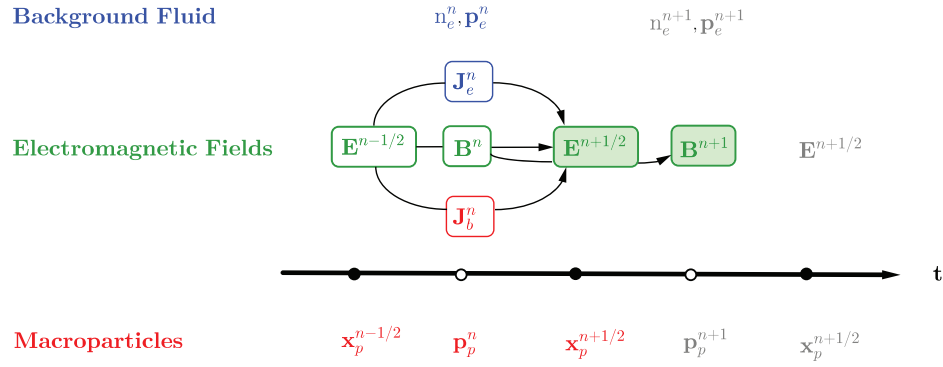

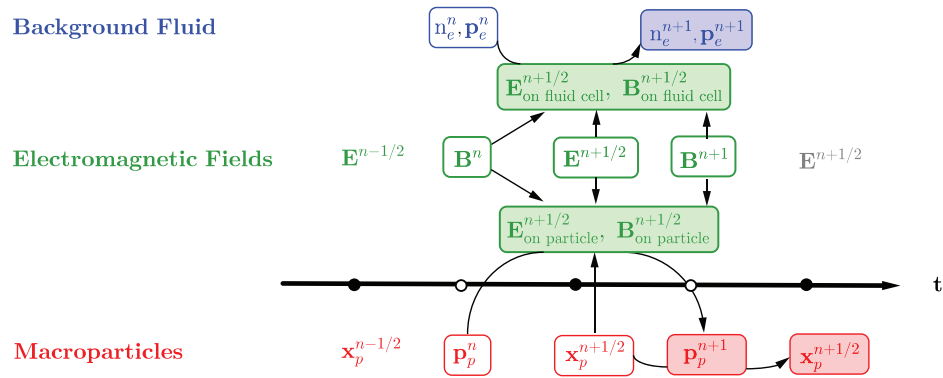Figure 6: Same as Fig.5, but showing electromagnetic fields advancement scheme.

Figure 7: Same as Fig.5, but showing the advancement of the particle and the fluid quantities.

3D-Gaussian bunch $\widetilde{Q} = \alpha(2\pi)^{3/2}\sigma_x\sigma_y\sigma_y k_p^3$, where σ_i $(i = x, y, z)$ represents the rms-size in the correspondent direction.

The test case chosen for the comparison is the propagation of a Gaussian electron bunch in a uniform plasma channel of unperturbed density $n_0 = 10^{16}$ cm^{-3}. The bunch parameters are: longitudinal rms-length $\sigma_z = 50$ μm, transverse rms-length $\sigma_x = \sigma_y = 9$ μm and relativistic factor $\gamma = 200$. To better compare the two codes, the same grid resolution has been used: the mesh cell size is 1 μm longitudinally, 0.4 μm transversely. Architect discretises the bunch with 15k particles, while ALaDyn, being 3D, uses 1.5 million particles. The background plasma in ALaDyn has been discretised with 8 particles per cell. The Architect simulation domain has been chosen with 460 cells in the longitudinal direction z and 240 cells in the transverse direction r. The ALaDyn simulation domain has been chosen with the same number of cells in the longitudinal direction z and 480 cells in each of the transverse directions x, y; the number of mesh points in each of the transverse directions in the ALaDyn simulation domain has been chosen as twice the number of transverse mesh points of the Architect simulation domain, since the latter includes only half plane in the transverse direction. The integration timesteps are $\Delta t = 0.77$ fs for ALaDyn, while $\Delta t = 0.67$ fs for Architect. The timestep for the hybrid simulation has been set to a slightly smaller value with respect to the full PIC simulation, to correctly catch with a fluid model the local increase of plasma oscillation frequency in the tail of the electron bubble in the driver's wake. Due to the hybrid code speed with respect to the full PIC simulation, the smaller timestep does not result in a detrimental decrease of the hybrid model speedup.

The two codes have been compared for four different regimes after 1 mm of propagation within the plasma, distance at which the system has reached a stationary behaviour. The considered normalized charge values are Q = [60, 120, 180, 240] pC, corresponding to $\widetilde{Q} = [0.25, 0.5, 0.75, 1.0]$. Such values of Q span the transition to highly nonlinear regimes, occurring at $\widetilde{Q} = 1.0$

A quantity of primary importance for PWFA is the accelerating field in the wake of the driver. A comparison for the different regimes of the electric field on-axis calculated by the codes is plotted in Fig. 8. We observe that for $\widetilde{Q} = [0.25, 0.5]$, the results by ALaDyn and Architect are essentially the same. With increasing $\widetilde{Q}$ the hybrid approach responds slightly differently than a PIC code. We observe, from Fig. 8, that discrepancies are concentrated at the tail where the bubble closes up. The cold fluid approximation cannot catch the charge concentration and particle overshooting at the end of the bubble and thus in a fluid model the field does not peak as much as in a PIC code. At $\widetilde{Q} = 0.75$ the maximum error in the electric field on axis is 30%, localized in a very narrow region where the field peaks. At $\widetilde{Q} = 1$. the length of the bubble is slightly different and the peak electric field is different by a 30%. However, the order of magnitude of the accelerating field can be assessed through Architect hybrid model. Similar differences in the rear of the bubble in the driver's wake were shown in highly nonlinear regimes in [32, 33], although they were related to features of the particular model presented in those references.

The transverse electric field has an essential role in the study of the transverse dynamics of a witness bunch injected in the driver's wake - i.e. the evolution of its emittance and its transverse size. Figure 9 shows a comparison of the transverse field computed with Architect and ALaDyn in a transverse section at distance $0.5\,\lambda_p$ behind the driver for $\widetilde{Q} = [0.75, 1.0]$.
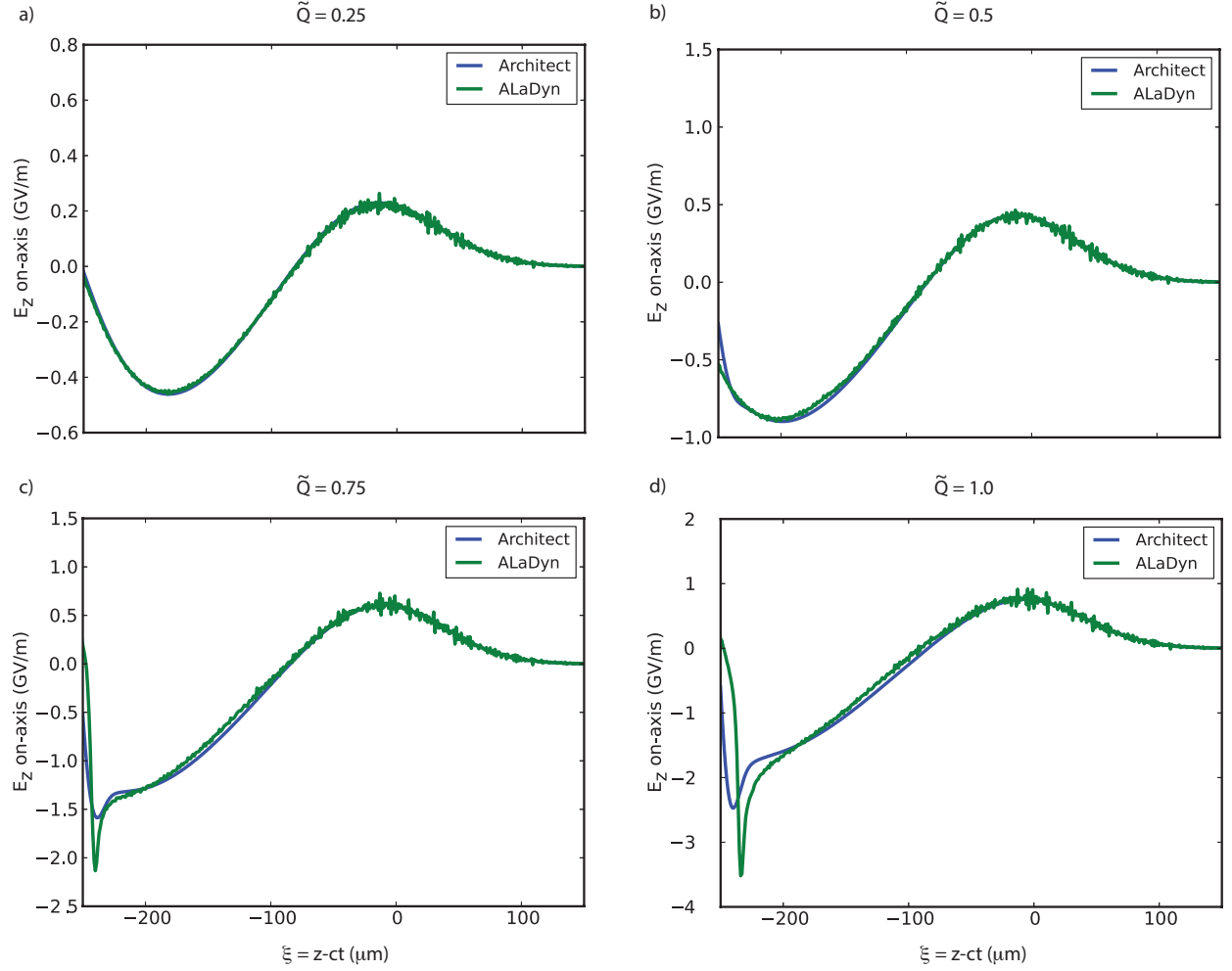
9

Figure 8: Panel a): Longitudinal electric field E_z on beam propagation axis - the bunch is moving from left to right; comparison between Architect and ALaDyn at 1 mm of propagation in the plasma, $\widetilde{Q} = 0.25$. Panel b): Same as Panel a), but with $\widetilde{Q} = 0.5$. Panel c): Same as Panel a), but with $\widetilde{Q} = 0.75$. Panel d): Same as Panel a), but with $\widetilde{Q} = 1.0$.

This position in the driver's wake represents a usual choice for witness positioning. The hybrid model well reproduces the PIC transverse field. Figure 10 shows a contour plot of the transverse electric field in the simulations with $\widetilde{Q} = [0.75, 1.]$. The overall field features, i.e. magnitude, sign, spatial variations, are well reproduced in the bubble region, where the acceleration and focusing of a witness bunch is performed in typical PWFA experiments.

A two dimensional density contour plot of the density is reported in Fig. 11, for the two cases of interest $\widetilde{Q} = [0.75, 1.]$. We observe that the basic features of the electron bubble are reproduced: length and bubble transverse dimension. We notice that at $\widetilde{Q} = 1$, Fig. 11 b), the electron accumulation layer at the bubble boundaries, well visible from ALaDyn simulations, it is also reproduced by the hybrid code in a smoother nature.

We remark that the shown ALaDyn simulations needed 4000 hours-core, while the shown Architect simulations needed only 1 hour on a single cpu. Despite the small computational time needed, the reduced model of Architect accurately reproduces the quantities of interest for PWFA - i.e. accelerating and focusing fields. Significant differences with a PIC code start to appear in highly nonlinear regimes ($\widetilde{Q} \geq 1.0$). Architect thus represents an effective tool for preliminary assessments up to weakly nonlinear regimes, that are of interest for the upcoming SPARC_LAB experiments.

5. Conclusions

We have presented Architect, a time-explicit hybrid kinetic-fluid code for electron PWFA; to validate the code and determine its range of validity we have compared Architect with the full PIC code ALaDyn.

We have described the code underneath algorithms, describing how the electron bunch(es) are treated kinetically with PIC techniques while the background plasma is modelled as a cold fluid. Bunches evolve in a 6D phase space. Hydrodynamic and Maxwell's equations are instead solved assuming cylindrical symmetry, in order to reduce computational costs. The beam-plasma system evolution is seeded and driven by the current term in Ampere's equation; such current is calculated as the superposition of the bunch current and the background current.

Bunch particles are evolved with a classical Boris' pusher, while the cold fluid quantities are evolved with a classical upwind scheme and the Maxwell's equations with an FDTD scheme. Each equation subset has been solved with a choice of a well-known scheme; time-space current centering as well as bunch self consistent field initialisation are implemented with special care.

Architect is a serial code and can be made run on a single core, the use of cylindrical symmetry combined with a fluid background allows to drastically reduce run times. For the tests proposed in the article the required time is 1 hour on a single core for a propagation of 1 mm in the plasma, a very competitive runtime for a time-explicit code. The hybrid strategy seems to represent a possible way towards PWFA systematic scans in the considered regimes.

As a final aspect, Architect has been compared versus a full 3D PIC code. We have observed the very accurate behavior of the code in the weakly non linear regime. The most challenging test compares ALaDyn and Architect in the transition to a non-linear regime; we notice that the hybrid code results begin to disagree in this case with the PIC results,
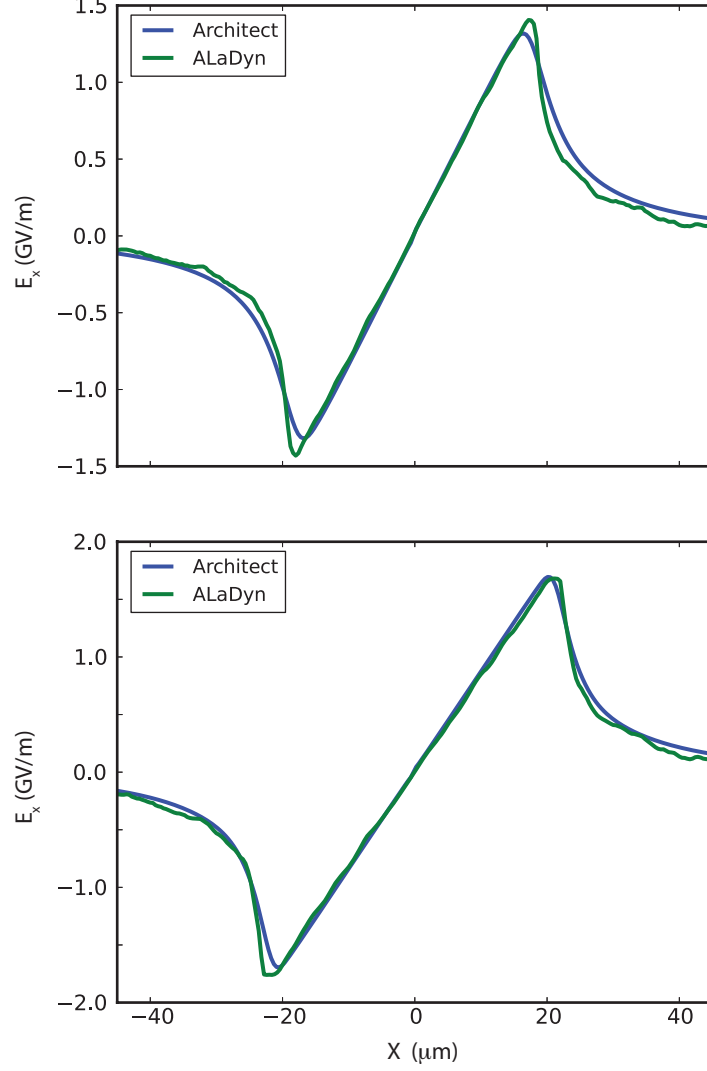
Figure 9: Top panel: transverse electric field E_x for $\widetilde{Q} = 0.75$ in a section perpendicular to the z axis, at distance $0.5 \lambda_p$ behind the driver, comparison between Architect (for which the transverse field is E_r) and ALaDyn at 1 mm of propagation in the plasma. The horizontal axis represents the x axis for ALaDyn and $[-R, R]$ for Architect. The Architect field in the zone $-R$ has been obtained by simmetry. Bottom panel: same as top panel but with $\widetilde{Q} = 1.0$
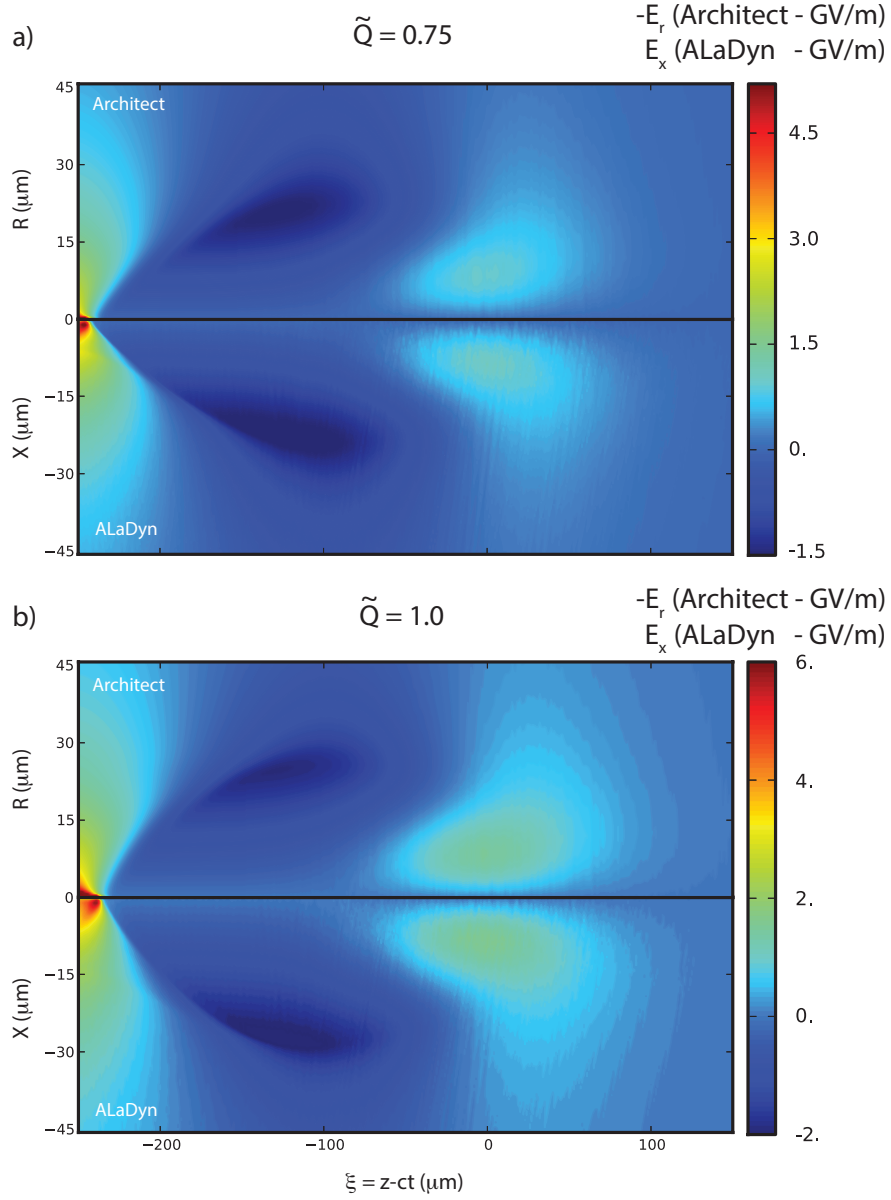
.

Figure 10: Panel a): 2D section of transverse electric field at 1 mm of propagation in the plasma, the bunch is moving from left to right; comparison between Architect (top half-panel) and ALaDyn (bottom half-panel), $\widetilde{Q} = 0.75$. Panel b): Same as panel a), but $\widetilde{Q} = 1.0$.
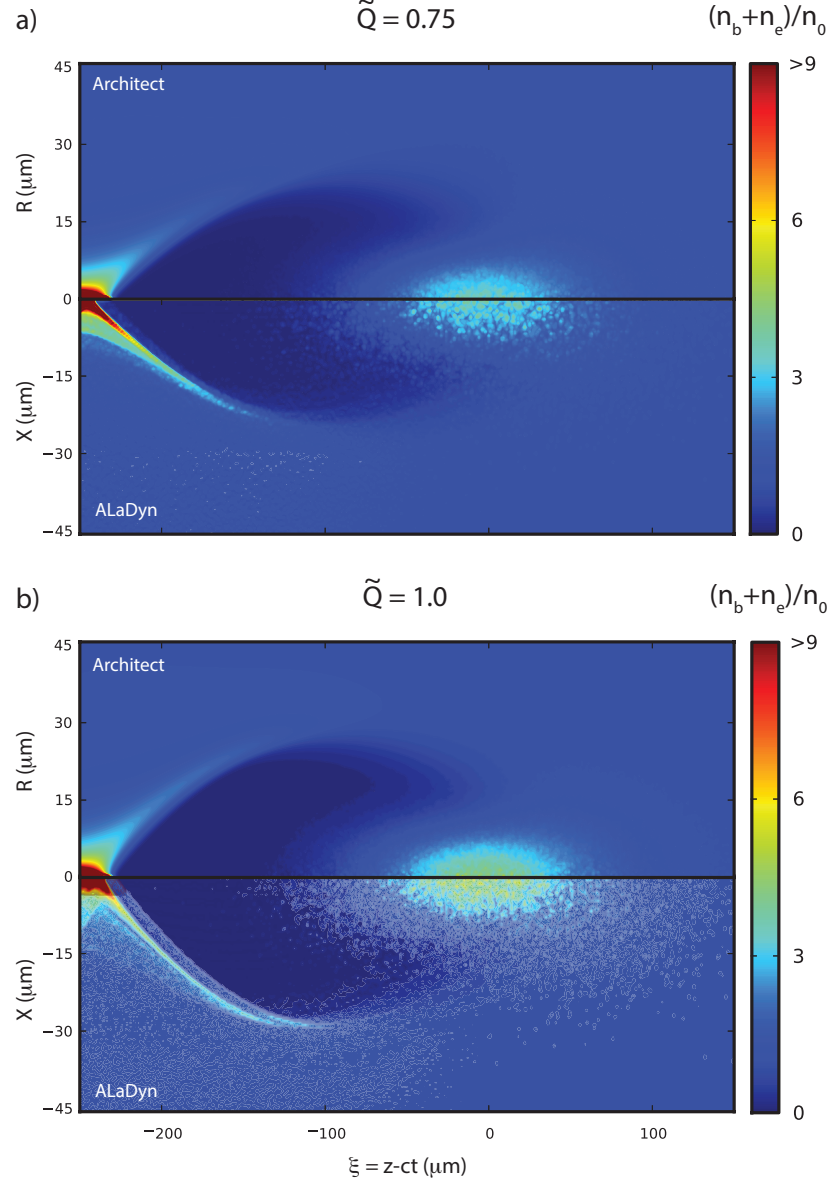
13

Figure 11: Panel a): 2D section of electron densiy (bunch+background) field at 1 mm of propagation in the plasma, the bunch is moving from left to right; comparison between Architect (top half-panel) and ALaDyn (bottom half-panel), $\widetilde{Q} = 0.75$. Panel b): Same as panel a), but $\widetilde{Q} = 1.0$.

14

the disagreement being localised at the rear of the bubble where the bubble closes up and a kinetic description is necessary.

Along the same investigation line we will be interested in comparing the reduced model code versus the full three dimensional PIC code for much longer distances so to identify where three dimensional effects and head erosion effects begin to play some role.

6. Aknowledgement

[1] T. Tajima and J. M. Dawson, Phys. Rev. Lett. **43**, 267 (1979).

[2] I. Blumenfeld, C. E. Clayton, F. J. Decker, M. J. Hogan, C. Huang, R. Ischebeck, R. Iverson, C. Joshi, T. Katsouleas, N. Kirby, W. Lu, K. A. Marsh, W. B. Mori, P. Muggli, E. Oz, R. H. Siemann, D. Walz, M. Zhou, Nature **445**, 741 (2007).

[3] M. Litos, E. Adli, W. An, C. I. Clarke, C. E. Clayton, S. Corde, J. P. Delahaye, R. J. England, A. S. Fisher, J. Frederico, S. Gessner, S. Z. Green, M. J. Hogan, C. Joshi, W. Lu, K. A. Marsh, W. B. Mori, P. Muggli, N. Vafaei-Najafabadi, D. Walz, G. White, Z. Wu, V. Yakimenko, G. Yocky, Nature **515**, 92 (2014).

[4] C. K. Birdsall, A. B. Langdon, Plasma Physics Via Computer Simulation CRC Press, Boca Raton (2004).

[5] C. Benedetti, C. B. Schroeder, E. Esarey, C. G. R. Geddes, W. P. Leemans, AIP Conf. Proc. **1299**, 250 (2010).

[6] K. V. Lotov, Phys. Plasmas **5**, 785 (1998).

[7] P. Mora, T. M. Antonsen, Phys. Rev. E **53**, R2068 (1996).

[8] D. H. Whittum, Phys. Plasmas **4**, 1154 (1997).

[9] P. Sprangle, E. Esarey, A. Ting, Phys. Rev. A **41**, 4463 (1990).

[10] K. V. Lotov, Phys. Rev. ST Accel. Beams **6**, 061301 (2003).

[11] B. Feng, C. Huang, V. Decyk, W.B. Mori, P. Muggli, T. Katsouleas, J. Comput. Phys. **228**, 5340 (2009).

[12] T. Mehrling, C. Benedetti, C. B. Schroeder, J. Osterhoff, Plasma Physics and Controlled Fusion **56**, 084012 (2014).

[13] J. B. Rosenzweig, N. Barov, M. Thompson, R. Yoder, Phys. Rev. ST Accel. Beams. **7**, 061302 (2004).

[14] P. Londrillo, C. Gatti, M. Ferrario, Nuclear Inst. and Methods in Physics Research, A **740**, 236 (2014).

[15] A. Marocchino, F. Massimo, http://dx.doi.org/10.5281/zenodo.49572 (2016).

[16] F. Massimo, A. Marocchino, A. R. Rossi, Nuclear Instruments and Methods in Physics Research A (2016), http://dx.doi.org/10.1016/j.nima.2016.02.043

[17] A. Marocchino, F. Massimo, A. R. Rossi, E. Chiadroni, M. Ferrario, Nuclear Instruments and Methods in Physics Research A (2016), http://dx.doi.org/10.1016/j.nima.2016.03.005

[18] S. Sinigardi, A. Marocchino, P. Londrillo, A. Sgattoni, http://dx.doi.org/10.5281/zenodo.49553 (2016).

[19] C. Benedetti, A. Sgattoni, G. Turchetti and P. Londrillo, IEEE Transactions on Plasma Science **36**, 1790 (2008).

[20] P. Londrillo, C. Benedetti, A. Sgattoni, G. Turchetti, Nucl. Instrum. Meth. A **620**, 1 (2010).

[21] M. Ferrario, D. Alesini, M.P. Anania, A. Bacci, M. Bellaveglia, O. Bogdanov, R. Boni, M. Castellano, E. Chiadroni, A. Cianchi, S.B. Dabagov, C. De Martinis, D. Di Giovenale, G. Di Pirro, U. Dosselli, A. Drago, A. Esposito, R. Faccini, A. Gallo, M. Gambaccini, C. Gatti, G. Gatti, A. Ghigo, D. Giulietti, A. Ligidov, P. Londrillo, S. Lupi, A. Mostacci, E. Pace, L. Palumbo, V. Petrillo, R. Pompili, A.R. Rossi, L. Serafini, B. Spataro, P. Tomassini, G. Turchetti, C. Vaccarezza, F. Villa, G. Dattoli, E. Di Palma, L. Giannessi, A. Petralia, C. Ronsivalle, I.P. Spassovsky, V. Surrenti, L. Gizzi, L. Labate, T. Levato, J.V. Rau, Nucl. Instrum. Methods Phys. Res. Sect. B, **309**,183 (2013).

[22] R. Pompili, M.P. Anania, M. Bellaveglia, A. Biagioni, F. Bisesto, E. Chiadroni, A. Cianchi, M. Croia, A. Curcio, D. Di Giovenale, M. Ferrario, F. Filippi, M. Galletti, A. Gallo, A. Giribono, W. Li, A. Marocchino, A. Mostacci, M. Petrarca, V. Petrillo, G. Di Pirro, S. Romeo, A.R. Rossi, J. Scifo, V. Shpakov, C. Vaccarezza, F. Villa, J. Zhu, Nuclear Instruments and Methods in Physics Research A, http://dx.doi.org/10.1016/j.nima.2016.01.061i (2016).

[23] B. M. Cowan, D. L. Bruhwiler, J. R. Cary, E. Cormier-Michel, C. G. R. Geddes, Phys. Rev. ST Accel. Beams **16**, 4 (2013).

[24] G. Pöplau, U. van Rienen, S. B. van der Geer, M. J. de Loos, Proceedings of EPAC 2002, Paris, France.

[25] J. Qiang, P. L. Colestock, D. Gilpatrick, H.V. Smith, T. P.Wangler, M. E. Schulze, Phys. Rev. Phys. Special Topics - Acc. and Beams **5**, 124201 (2002).

[26] K. Yee, IEEE Transactions on Antennas and Propagation **14**, 3 (1966).

[27] L. Yongdong, H. Feng, L. Chunliang, Plasma Science and Technology **7**, 1 (2005).

[28] J. P. Boris, D. L. Book, J. Comput. Phys. **11**, 38 (1973).

[29] S. T. Zalesak, J. Comput. Phys. **31**, 335 (1979).

[30] F. Massimo, A. Marocchino, E. Chiadroni, M. Ferrario, A. Mostacci, P. Musumeci, L. Palumbo, Nuclear Inst. and Methods in Physics Research, A **740**, 242 (2014).

[31] J. B. Rosenzweig, G. Andonian, M. Ferrario, P. Muggli, O. Williams, V. Yakimenko, and K. Xuan, AIP Conference Proceedings **1299**, 500 (2010).

[32] W. Lu, C. Huang, M. M. Zhou, W. B. Mori, T. Katsouleas, Phys. Rev. Lett. **96**, 165002 (2006).

[33] W. Lu, C. Huang, M. Zhou, M. Tzoufras, F. S. Tsung, W. B. Mori, and T. Katsouleas, Physics Plasmas **13**, 056709 (2006).