



# Application of adaptive mesh refinement to PIC simulations in heavy ion fusion<sup>☆</sup>

J.-L. Vay<sup>a,b,\*</sup>, A. Friedman<sup>a,c</sup>, D.P. Grote<sup>a,c</sup>

<sup>a</sup>Heavy Ion Fusion Virtual National Laboratory, USA

<sup>b</sup>Lawrence Berkeley National Laboratory, Bldg. 47/112, 1 Cyclotron Road, Berkeley, CA 94720, USA

<sup>c</sup>Lawrence Livermore National Laboratory, CA, USA

## Abstract

An end-to-end simulation of a Heavy-Ion Fusion driver is very challenging due to the wide range of scales involved. We have merged the Particle-In-Cell and the Adaptive Mesh Refinement (AMR) methods and have applied it to the simulation of HIF beam sources. We describe our past progress and present our recent implementation of automatic AMR in WARP.

© 2005 Published by Elsevier B.V.

PACS: 52.65.Rr; 52.58.Hm; 41.75.Ak; 02.70.Bf

Keywords: Mesh refinement; Particle-in-cell; Heavy-ion fusion; Plasma simulation; Charged particle beam simulation

## 1. Introduction

Plasma simulations are often rendered challenging by the disparity of scales in time and in space which must be resolved. This is true, for example, for the modeling in detail of the beam behavior from end-to-end in a Heavy Ion Fusion accelerator. Although such a simulation is ultimately

needed, a first principles treatment is at present beyond reach, mostly due to the fact that the space scales span up to nine orders of magnitude, from microns (the Debye Length close to the emitter) to a kilometer or more (the length of the accelerator). The mesh refinement (MR) technique addresses the issue of a wide range of space scales and has been applied successfully in other fields (e.g., fluid dynamics). While preserving most of the advantages of regular gridding, this technique covers areas that need a higher resolution with a finer mesh. If the areas of the physical domain that need refinement evolve in time, then an automatic redistribution of the refinement following some

<sup>☆</sup>Work performed for USDOE under Contracts DE-AC03-76F00098 at UC-LBNL and W-7405-ENG-48 at UC-LLNL.

\*Corresponding author. Lawrence Berkeley National Laboratory, Bldg. 47/112, 1 Cyclotron Road, Berkeley, CA 94720, USA. Fax: +1 510 486 5392.

E-mail address: [jlway@lbl.gov](mailto:jlway@lbl.gov) (J.-L. Vay).

criterion applies: this is known as adaptive mesh refinement (AMR). In Ref. [1], we discussed the issues associated with merging the AMR method with PIC for both the electrostatic and the electromagnetic cases, including spurious self-forces in both cases and spurious reflection of waves with amplification in the case of electromagnetics. We also presented an example of simulation of an ion source (triode) which demonstrated the potential benefits of MR for PIC. It was shown that the longitudinal emittance profile obtained by a run at fine resolution could be recovered by a less costly simulation using a fixed MR patch surrounding the emitting area.

## 2. Summary of achievements since the last Symposium

In Ref. [2], further analysis of the study reported in Ref. [1], using additional diagnostic (charge density profiles at the exit of the triode) revealed that, although the longitudinal emittance profile was well recovered, the edge of the beam distribution was not accurately represented. We presented in the same paper a new simulation of the case studied in Ref. [1] which used a mock-up of an AMR patch wherein the beam edge was refined, in addition to the emitting area. The benefits on the beam edge distribution were obvious on the phase-space and charge density profile plots. Also in the same paper, a one-dimensional study of constraints on the rapidity of the beam current rise time showed that MR was an essential tool for getting numerical convergence with reasonable resources. A new kind of refinement patch, specialized for the emitting region of a source emitting in the Child-Langmuir limit, was introduced. Its uniqueness lies in its variable cell size set to match the inverse of the charge density at the Child-Langmuir limit. The study also showed that an additional adaptive refinement patch that follows the head of the beam was needed to avoid a spike of density formed by particles being given a spuriously low acceleration because of a lack of grid resolution. The new method was implemented in a one-dimensional solver in WARP and was validated against an analytical solution. A more

detailed presentation of the implementation of MR patches into WARP-*rz* was reported in Ref. [3]. The structure of the data and the methods for depositing the charge and gathering forces were given in that paper. In Ref. [4], we introduced a new method for the application of MR in an electromagnetic PIC code, implemented in an existing 2-D electromagnetic PIC code developed and used at Ecole Polytechnique (France) for laser-plasma interaction studies. Results showed acceptable performance and confirmed the stability of the method. In Ref. [5], an extended summary of the results briefly summarized here is given, as well as an example of application, in 2-D axisymmetric geometry, of the specialized refinement patch developed in one dimension and introduced in Ref. [2]. Comparisons with experiments showed good agreement and provided an experimental validation of the method. In addition, progress of the authors' collaboration with the Applied Numerical Algorithms Group (ANAG) at LBNL, to upgrade ANAG's Chombo MR library with all the tools necessary for providing MR capability to existing PIC codes, have been reported in Refs. [1–3,5,6].

## 3. Implementation of automatic AMR in WARP-*rz*

Automatic generation of MR patches has recently been implemented in WARP. Given the charge density  $\rho$  on a  $N_r \times N_z$  array, a list of patches is generated according to the absolute values of the charge density and its gradient. The generation of the list is a two-step process. First, a  $N_r \times N_z$  array containing the required level of refinement at a grid node level is generated. Then, nodes are coalesced into patches and a list of patches is produced. Both processes are described in detail in the Appendix. For accuracy, a patch can have only a parent-child relationship with patches at the refinement levels immediately above or below its own level. This translates into the fact that one node tagged for a refinement level  $l$  is by definition tagged for all the intermediate levels between 1 (the level of the main coarse grid) and  $l$ . This ensures that there is always a smooth transition from one level to the next, and so

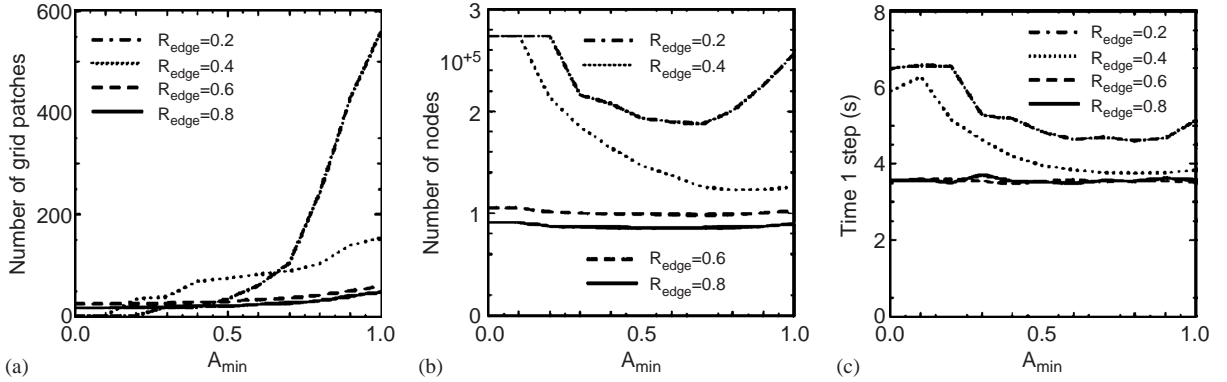


Fig. 1. (a) Number of grid patches, (b) number of nodes and (c) timing for one step as a function of  $A_{\min}$  for  $R_{\text{edge}} = (0.2, 0.4, 0.6, 0.8)$ .

reduces errors in the interpolation of the values on the boundaries of a patch from a coarser level. In the current implementation, gradients are refined at a given refinement level, e.g. there is no gradual refinement based on the magnitude of the gradient. This explains what appear to be “double patches” in Fig. 2. The refinement following the gradient of the charge density (computed on an auxiliary array) is controlled by a parameter,  $0 \leq R_{\text{edge}} \leq 1$ , which specifies the threshold, relative to the maximum, of the magnitude of the gradient above which the refinement applies (this is done on a row and column basis, see Appendix). A value of  $R_{\text{edge}} = 0$  or  $R_{\text{edge}} = 1$  will respectively result in refinement everywhere or only at the location of the maximum of the gradient.

The coalescence step is an optimization process targeted at finding an optimum between the total number of nodes in all the patches and the number of patches. During the coalescence process, nodes that have been flagged at different refinement levels might be forced to coalesce together (since a patch is always rectangular). A patch pertains to a specific refinement level, say  $l$ . Thus, for such a patch, the nodes that have been tagged at the refinement level  $l$  are flagged as “active” while the others are flagged as “inactive”. Both inactive and active nodes are used for charge deposition and field solution, but only the active nodes are used for force gathering. This ensures that the effective shape of the patch matches exactly the collection of nodes tagged at the refinement level  $l$  only. We

denote the ensemble of active nodes in a patch as the “effective” area of the patch and define the ratio of the number of active nodes divided by the number of nodes in a patch as its relative effective area. The input parameter  $A_{\min}$  fixes the minimum value authorized for the relative effective area of any patch. When forming a patch, rows or columns of nodes are allowed to join an existing patch only if the resulting relative effective area is larger than, or equal to,  $A_{\min}$ . A low value of  $A_{\min}$  (close to 0) will produce aggressive coalescence and might result in too many inactive nodes. On the other hand, a high value (close to 1) will produce very little coalescence and might result in a very large number of patches, where the overhead associated with patch management might become excessive. Another caveat of setting  $A_{\min}$  close to 1 is that each patch is surrounded by layers of transition nodes to reduce the effect of spurious self-forces [1]. More patches implies more transition nodes.

The number of grid patches, total number of nodes (over all patches) and the timing for one time step are given in Fig. 1 as functions of the relative effective area  $A_{\min}$  for four values of  $R_{\text{edge}}$  (0.2, 0.4, 0.6, 0.8). As expected, the number of grid patches rises with  $A_{\min}$  and  $R_{\text{edge}}$ , as does the total number of grid nodes. However the total number of grid nodes is constant for  $0.6 \leq R_{\text{edge}} \leq 0.8$ , decreasing for  $R_{\text{edge}} = 0.4$  and decreasing before increasing for  $R_{\text{edge}} = 0.2$ . The decrease for the case  $R_{\text{edge}} = 0.2$  and  $R_{\text{edge}} = 0.4$  is attributed to a

decrease of nodes populating the inactive area of the patches. The following increase for the case  $R_{\text{edge}} = 0.2$  is attributed to the increase in transition nodes. The reason for the nearly flat curves for the cases  $R_{\text{edge}} = 0.6$  and  $R_{\text{edge}} = 0.8$  is easier to understand by analysing the plots of the charge density and the grid patches boundaries in Fig. 2 for the cases  $A_{\text{min}} = (0., 0.5, 0.8, 1.)$  for  $R_{\text{edge}} = (0.4, 0.6)$ . The case  $R_{\text{edge}} = 0.4$  is as expected, with the entire distribution being enclosed into one large patch for  $A_{\text{min}} = 0$  and many

small patches covering the emitter, the beam edge and density fluctuations on axis for  $A_{\text{min}} = 1$ . For  $R_{\text{edge}} = 0.6$ , small parts of the beam edge had gradients below 60% of gradients due to density fluctuations on axis and were not eligible for refinement.

This created gaps between refinement patches on the edge which prevented them from coalescing, resulting in a constant set of patches following the beam edge for  $0 < A_{\text{min}} < 1$ . Only the emitter region was covered by a single patch at  $A_{\text{min}} = 0$  and

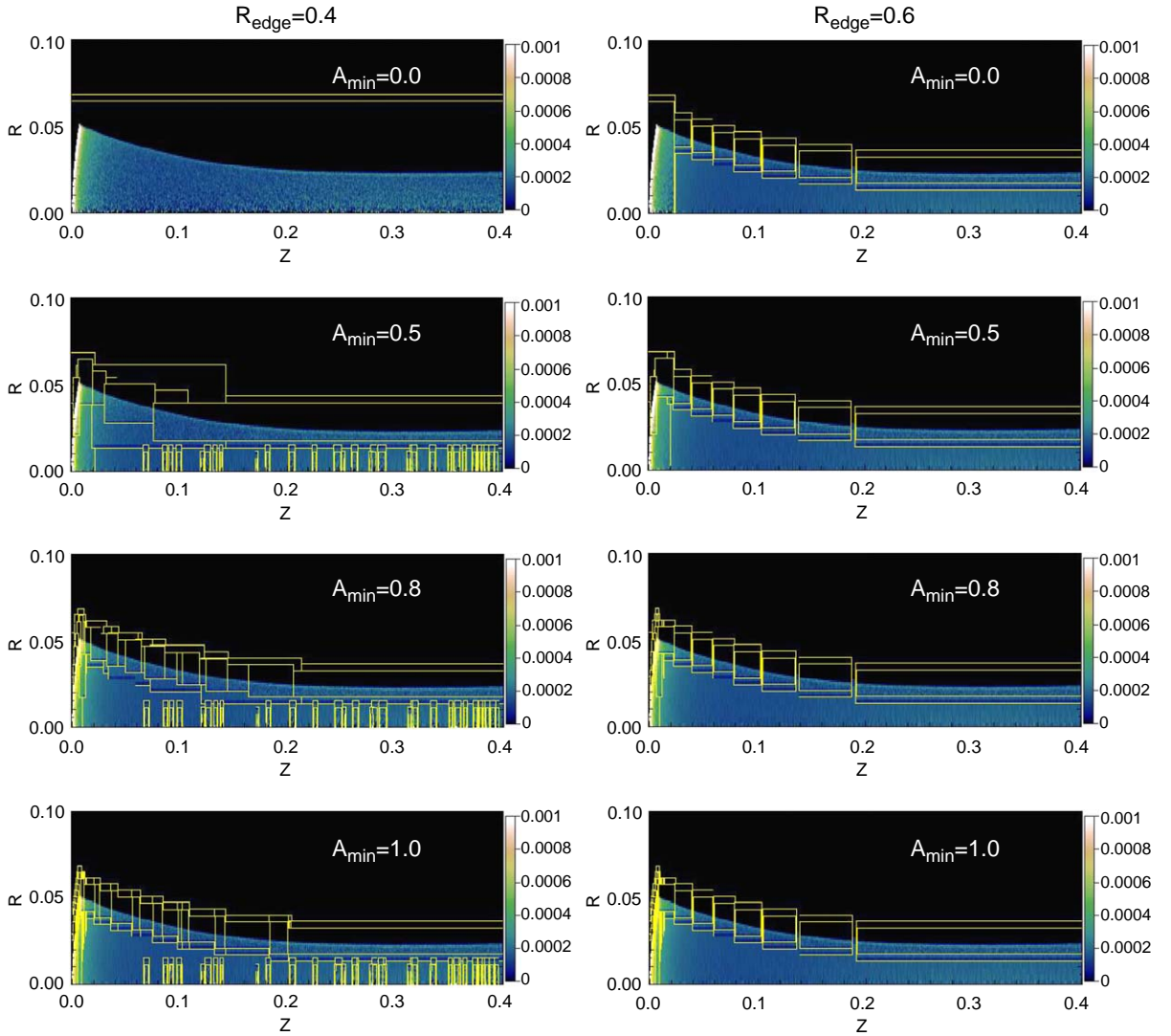


Fig. 2. Charge density and grid patches boundaries for the cases  $A_{\text{min}} = (0, 0.5, 0.8, 1.)$  and  $R_{\text{edge}} = (0.4, 0.6)$ .

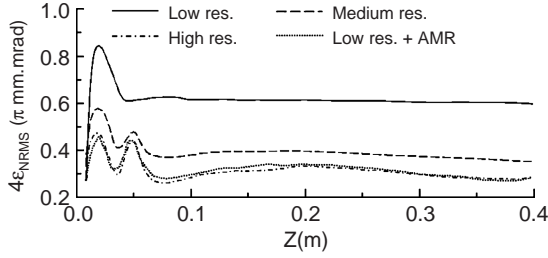


Fig. 3. Beam rms emittance as a function of  $z$  (statistical noise was removed for clarity): the emittance converges downward with increasing resolution. The high resolution result is recovered with a run at low resolution and AMR.

many patches at  $A_{\min} = 1$ . The timing for one step closely follows the number of nodes, but does not seem to depend on the number of patches, even in the case ( $R_{\text{edge}} = 0.2$ ,  $A_{\min} = 1$ ) where about 550 grid patches were created, indicating that the overhead associated with patch management is very small. From this, we conclude that the range of optimum value for  $A_{\min}$  lies between 0.6 and 0.9. Note that the maximum number of macro-particles in the simulation was around 1 million, thus from 10 to 4 times the maximum number of nodes, which is quite a typical number for PIC calculations. We expect that going to larger particle numbers will diminish the relative cost of finer meshes (since the particles will increasingly dominate the cost). A test performed with  $R_{\text{edge}} = 0.4$  and  $A_{\min} = 0.8$  recovered the emittance profile presented in Refs. [1,2] at high resolution, as can be seen in Fig. 3, with a speed-up factor of about 11.3 compared to a non-refined fine grid calculation.

#### 4. Conclusion

We have merged the Particle-In-Cell and the Adaptive Mesh Refinement methods and have completed a full featured implementation in WARP- $rz$ . Results reported elsewhere have demonstrated the effectiveness of the new capability. We have recently added capabilities for automatic generation of AMR patches and have performed a preliminary analysis of efficiency as a function of number of grids and nodes generated as well as the

proportion of active versus inactive nodes. The new capability has allowed us to recover previously published results at a lower computational cost than was previously achievable.

#### Appendix

The procedure used to generate the array of refinement levels  $L_{\text{dens}}$  from the charge density  $\rho$  is, defining  $n$  to be the number of refinement levels and  $r$  to be the refinement ratio between two levels:

- (1)  $L_{\text{tmp}} = \max(1, r^{2(n+1)} \rho / \max(\rho))$ ,
- (2)  $L_{\text{dens}} = \min(r, \text{int}(\log(L_{\text{tmp}}) / \log(r^2)))$ .

The procedure used to generate the array of refinement levels  $L_{\text{grad}}$  from the charge density gradient, defining  $l_{\text{grad}}$  to be the refinement level and  $0 \leq R_{\text{edge}} \leq 1$ , is

- (1)  $L_{\text{tmp}} = 0.5(|\vec{\nabla}_x \rho| + |\vec{\nabla}_y \rho|)$ ,
- (2) for each row (we denote  $L_{\text{tmp}}^r$ ) of  $L_{\text{tmp}}$ :  
 $L_{\text{grad}}^r = L_{\text{tmp}}^r$  where  $L_{\text{tmp}}^r > R_{\text{edge}} * \max(L_{\text{tmp}}^r)$  and  $L_{\text{grad}}^r = 0$  otherwise,
- (3) for each column (we denote  $L_{\text{tmp}}^c$ ) of  $L_{\text{tmp}}$ :  
 $L_{\text{grad}}^c = L_{\text{tmp}}^c$  where  $L_{\text{tmp}}^c > R_{\text{edge}} * \max(L_{\text{tmp}}^c)$  and  $L_{\text{grad}}^c = 0$  otherwise,
- (4)  $L_{\text{grad}} = l_{\text{grad}}$  where  $\max(L_{\text{grad}}^r, L_{\text{grad}}^c) \neq 0$  and  $L_{\text{grad}} = 1$  otherwise.

The final array of refinement levels  $L$  is obtained from  $L = \max(L_{\text{dens}}, L_{\text{grad}})$ . The coalescence process is done for each refinement level  $l$  as follows, with  $0 \leq A_{\min} \leq 1$  the effective area (explained below):

- (1) the first node in the list of nodes tagged at refinement level  $l$  is upgraded to patch status,
- (2) successive tentative extensions of the patch, in  $z$  in the forward direction (east), in  $r$  in the backward direction (south), in  $z$  in the backward direction (west) and in  $r$  in the forward direction (north) are performed. For each tentative extension, a ratio  $a$  counting the number of nodes where  $L \geq l$  divided by the number of nodes in the tentative patch is computed. If  $a \geq A_{\min}$  and if no node in the

tentative patch has already been claimed by another patch at the same refinement level, the tentative extension is declared successful. Otherwise, the patch retains its initial boundaries (boundaries preceding the tentative extension),

- (3) step 2 is iterated until none of four successive (east, south, west, north) tentative extension is successful. Then the patch is added to the list of patches,
- (4) steps 1, 2 and 3 are iterated on the next available node at refinement level  $l$  not already claimed by a patch, until no node is left,
- (5) isolated patches totalizing four nodes or less, and thus corresponding (probably) to density fluctuations, are removed from the list of patches.

## References

- [1] J.-L. Vay, P. Colella, P. McCorquodale, B. Van Straalen, A. Friedman, D.P. Grote, *Laser Part. Beams* 20 (2002) 569.
- [2] J.-L. Vay, A. Friedman, D.P. Grote, *Proceedings of Seventh International Computational Accelerator Physics Conference*, East Lansing, IOP Publishing, Boston, New York, 2002.
- [3] J.-L. Vay, A. Friedman, D.P. Grote, P. Colella, P. McCorquodale, D.B. Serafini, *Comput. Phys. Commun.* 164 (1–3) (2004) 297.
- [4] J.-L. Vay, J.-C. Adam, A. Héron, *Comput. Phys. Commun.* 164 (1–3) (2004) 171.
- [5] J.-L. Vay, P. Colella, J.W. Kwan, P. McCorquodale, D.B. Serafini, A. Friedman, D.P. Grote, G. Westenskow, J.-C. Adam, A. Heron, I. Haber, *Phys. Plasmas* 11 (5) (2004) 2928.
- [6] P. McCorquodale, P. Colella, D.P. Grote, J.-L. Vay, J. *Comput. Phys.* 201 (1) (2004) 34.