C. R. Mecanique 342 (2014) 560–569

Contents lists available at ScienceDirect

Comptes Rendus Mecanique

www.sciencedirect.com

Theoretical and numerical approaches for Vlasov–Maxwell equations

Mathematical and numerical methods for Vlasov–Maxwell equations: The contribution of data mining

Franck Assous [a,b], Joël Chaskalovic [c,*]

[a] Ariel University, 40700 Ariel, Israel
[b] Bar-Ilan University, 52900 Ramat-Gan, Israel
[c] D'Alembert, University Pierre and Marie Curie, Paris, France

ARTICLE INFO

Article history:
Received 1 April 2014
Accepted 1 June 2014
Available online 6 August 2014

Keywords:
Data mining
Error estimate
Vlasov–Maxwell equations
Asymptotic analysis
Paraxial model

ABSTRACT

This paper deals with the applications of data mining techniques in the evaluation of numerical solutions of Vlasov–Maxwell models. This is part of the topic of characterizing the model and approximation errors via learning techniques. We give two examples of application. The first one aims at comparing two Vlasov–Maxwell approximate models. In the second one, a scheme based on data mining techniques is proposed to characterize the errors between a P_1 and a P_2 finite element Particle-In-Cell approach. Beyond these examples, this original approach should operate in all cases where intricate numerical simulations like for the Vlasov–Maxwell equations take a central part.

1. Introduction

There exist a lot of formulations that can model plasma physics or particle accelerators problems [1], as the Vlasov–Maxwell equations. Some of these formulations are very complete and complex, whereas some others are approximated, like the static, quasi-static or paraxial models [2–7]. In the same way, there exist a lot of possible numerical methods to discretize these models, like finite-element, finite-volume, Particle-In-Cell methods or semi-Lagrangian ones [8–12]. For all these mathematical and numerical methods, there exist error estimates and theoretical evaluations that characterize their relative accuracy. However, could we propose a more practical way to compare or evaluate these methods?

Our idea is to evaluate them by using data mining techniques, directly processed on the computed results. Even this approach is yet heuristic, it presents the advantage to allow one to work directly on the results, like a kind of *a posteriori analysis*. In this spirit, this can be compared to some recent approaches derived from reduced base method [13], in which these ideas are developed to efficient linear algebra methods.

The principle to extend the scope of data mining techniques to scientific computing was first introduced in [14]. We started from the observation that numerical methods produce today a huge quantity of numerical results, especially with the availability of massively parallel computers. Then, we noted that data mining techniques have already proved to be efficient in other contexts which deal with huge data, like in biology [15], medicine [16,17], marketing [18], advertising, and communications [19,20]. Hence, a first attempt at comparing asymptotic models with data mining was used in [14], whereas the idea to compare several sources of errors with data mining was stated in [21].

In this paper, we review this principle and focus on the contribution of data mining to mathematical and numerical models for Vlasov–Maxwell equations. Indeed, using data mining is particularly well adapted to a very "rich" model like the Vlasov–Maxwell one, for which:

[*] Corresponding author.

http://dx.doi.org/10.1016/j.crme.2014.06.010

1. the model is complete but complex and hard to accurately solve,
2. there are many approximate models, which are easier to solve numerically,
3. there are several different numerical methods with different levels of complexity and of precision.

For these reasons, this approach would probably not be well suited in a "quite simple" model for which there are not many different choices of modeling and approximations. It is well adapted to a real system modeled by a set of partial differential equations, the solution of which being carried out by numerical approximations methods. In these conditions, both modeling and approximation processes are different sources of errors (see [21]), that can in reality co-exist. The data mining approach proposed here could provide useful insights when difference types of errors are present, and could help to investigate possible interferences between these different types of errors.

For illustrative purpose, we will focus in this paper on two basic examples. We will first consider the system of Vlasov–Maxwell equations and first approximate it by a hierarchy of asymptotic models. We will then choose the first two simple models denoted M_1 and M_2 and propose to compare them *via* data mining techniques. More precisely, we try to understand in which way model M_2, which is *a priori* more precise, improves model M_1. This will requires the construction of a database made of all the numerical results computed by models M_1 and M_2.

The second example is devoted to the illustration of how data mining methods can allow us to evaluate error estimates between two approximations of the same model. More precisely, we consider a P_1 and a P_2 finite-element approximation of a Vlasov–Maxwell asymptotic model. A discretization error is defined as the error due to the difference of order between these two methods. Our aim is then to detect and characterize, by using data mining techniques, situations where using the P_2 finite-element method does not improve significantly the numerical results compared with the P_1 results.

This paper is organized as follows. In Section 2, we briefly present some notions and definitions of data mining that will be useful for the rest of the paper. In Section 3, a hierarchy of approximate Vlasov–Maxwell models will be derived. These will be the models used in our two fundamental illustrations. The first example of comparison between two asymptotic models is then exposed in Section 4, whereas Section 5 is devoted to the comparison of two finite-element methods. Perspectives will be drawn in the Conclusion. Let us emphasize that, beyond this study, this novel (even heuristic) approach could operate in other sources of errors, where simulations take a central part.

2. Principle of data mining exploration

Data mining is an activity of information extraction, whose goal is to discover hidden or *a priori* unknown facts contained in databases. Using a combination of machine learning, statistical analysis, modeling techniques and database technology, data mining finds patterns and subtle relationships in data and infers rules that allow the prediction of future results [22,23,18,24].

To make the article self-contained even for a reader not aware of these techniques, we briefly present now some notions and definitions, useful for the rest of the paper. The data mining technique used in what follows is the decision tree. Decision trees [25] belong to the *supervised data mining* methods to process segmentation. The purpose of segmentation is to constitute homogeneous subgroups inside a given population regarding a target variable which is to be explained *versus* predictor variables. This is processed by an algorithm of segmentation which is basically a minimization of the standard deviation for the concerned target variable.

In the case of the segmentation we will consider in our study, the target variable is a categorial one; It describes the belonging to one of several classes which characterizes a given level (called "Low", "High" and so on), of a target variable to be explained.

A decision tree is then composed by different subgroups (called *nodes*) of the initial population (called *root node*). These *nodes* are obtained with the segmentation algorithm by identifying among the predictor variables the most discriminating one regarding the homogeneity degree of the resulting *nodes*.

Each split of the segmentation divides a given *node* into several *nodes* (here, in our study into two nodes, which is the specific case of binary decision trees), based on the most discriminating predictor variable *var* such that the left resulting *node* obeys the inequality $var \leq \tau$ and the right one $var > \tau$ (τ being a threshold optimally computed by the algorithm of segmentation).

This process stops when the splitting is not feasible: either any new subgroup can be found to be more homogeneous than the previous one or the resulting segmentation is composed by insignificant subgroups. The path from the *root node* to each *leaf* (each terminal node) defines a succession of inequalities on the predictor variables that characterize the solutions belonging to the *leaf* with a certain risk which depends on the percentage of misclassified solutions in the *leaf*.

By choosing the *leaves* that predict the membership to the class of interest with the minimum risk, one is able to characterize this class with a set of "rules" at minimum risk.

3. A hierarchy of approximate Vlasov–Maxwell models

The first step of this study consists in introducing a hierarchy of approximate Vlasov–Maxwell models. We will then choose and compare two of them by data mining tools.

In order to define an approximate model, one has often to neglect one or several terms in the equations. The underlying idea is to identify parameters whose values can be small (and thus possibly negligible). To derive a hierarchy of approximate Vlasov–Maxwell models, one can perform an asymptotic analysis of those equations with respect to the parameters. This series of models is called a hierarchy, since considering a supplementary term in the asymptotic expansion leads to a new approximate model, theoretically aimed to be more precise. An analogous principle is used for instance for building approximate paraxial models of the propagation wave equation in geophysics [26,27].

From a numerical point of view, the approximate models are useful first and foremost if they coincide with a physical framework and second if they can efficiently solve the problem at a lower cost. In the sequel, we will consider the axisymmetric Vlasov–Maxwell equations, and show briefly how to build formally such approximate paraxial models.

So consider a beam of charged particles, for instance electrons, with a mass m and a charge q, which moves inside a perfectly conducting cylindrical tube of boundary $\varGamma$. Let us denote by z the axis of the tube, which is also the optical axis of the beam. Due to the axisymmetric features, we introduce the cylindrical coordinates (r, θ, z). The boundary $\varGamma$ is thus written $\varGamma = \{(r, \theta, z);\ r = R\}$, R being the radius of the tube. We also denote by $\boldsymbol{\nu}$ the unit outward normal to $\varGamma$. For the sake of simplicity, we assume here that there is no external field.

As for a non-collisional beam, the motion of these particles is described in terms of a particle distribution function $f(\mathbf{x}, \mathbf{p}, t)$ by the relativistic axisymmetric Vlasov equation (details can be found for instance in [14]).

The electromagnetic Lorentz force involved in the equation is expressed by $\mathbf{F} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B})$, where $\mathbf{F} = (F_r, F_\theta, F_z)$ describes how an electromagnetic field $\mathbf{E} = (E_r, E_\theta, E_z)$ and $\mathbf{B} = (B_r, B_\theta, B_z)$ acts on a particle that moves with a velocity $\mathbf{v} = (v_r, v_\theta, v_z)$. This electromagnetic field satisfies the classical axisymmetric Maxwell equations in the vacuum (see, for instance, [9]). The right-hand sides of Maxwell's equations are constituted by the charge density ρ and the current density $\mathbf{J} = (J_r, J_\theta, J_z)$, obtained as the zeroth and the first moments of the distribution function f solution to the Vlasov equation with

$$\rho = q \int f \, \mathrm{d}\mathbf{p}, \qquad \mathbf{J} = q \int \mathbf{v}(\mathbf{p}) f \, \mathrm{d}\mathbf{p} \tag{1}$$

The hierarchy of the paraxial models is then derived by exploiting the physical and geometrical properties of the problem, in the same spirit as in [2,5–7]. We briefly recall the principle here.

Let us assume that the beam—the dimension of which is small compared to the longitudinal length of the device—is highly relativistic, *i.e.* satisfies $\gamma \gg 1$, where γ denotes the Lorentz factor, $\gamma = (1 - \frac{\mathbf{v}^2}{c^2})^{-1/2}$. In fact, the notion of "highly relativistic" as well as the symbol "$\gg$" are rather vague, and need in practice to be more precisely defined or estimated. It is precisely one of the goal we want to achieve by using data mining techniques. It is thus convenient to rewrite the Vlasov–Maxwell equations in a frame, which moves along the z-axis with light velocity c. This requires to perform a change of variables by introducing the new position-velocity coordinates ζ and v_ζ defined by:

$$\zeta = ct - z, \qquad v_\zeta = c - v_z \tag{2}$$

We also assume that the longitudinal particle velocities v_z are close to the light velocity c, whereas the transverse particle velocities $(v_r^2 + v_\theta^2)^{1/2}$ are small compared to c. Hence, the particles appear to drift slowly in the direction $\zeta > 0$ in the new coordinates (r, ζ).

As a second step, we introduce a scaling of these equations based on characteristic values, from which we build dimensionless equations. Obviously, the choice of the scaling together with the dimensionless independent variables is crucial for the derivation of the approximate models. Different choices will give different approximate models, with or without the same theoretical precision, see for instance [28]. Here again, data mining methods can help to practically compare and evaluate these different approximate models. Then, introducing a small parameter, for instance in our case $\eta = \frac{\bar{v}}{c} \ll 1$, $\bar{v}$ denoting the particle transverse characteristic velocity, a hierarchy of approximate models is derived by retaining successively the first, second, third, etc. order in the asymptotic expansion of the distribution f. For instance, the paraxial models described respectively in [4] and [28] are different, but both are derived by retaining the terms up to the third order in the asymptotic expansion of the distribution function f. In the following section, we will compare the two first models obtained using this procedure.

4. Data mining to compare two approximate models

We consider here the two first models obtained by the procedure described above. We denote by M_1 the first-order model, corresponding to the asymptotic expansion of f up to the order 1, namely in which order-2 terms have been neglected. It is shown in [9,4] that this expansion is entirely determined from the knowledge of the 0-order expansion of the electromagnetic Lorentz force $(F_r^{M_1}, F_\theta^{M_1}, F_z^{M_1})$. To compute it, one proved that the electromagnetic field associated with the model M_1 (denoting ^{M_1}) has to satisfy the following equations,

$$\begin{cases} E_r^{M_1} = cB_\theta^{M_1} = \dfrac{1}{\varepsilon_0 r} \displaystyle\int_0^r \rho^{M_1} s \, \mathrm{d}s \\ E_\theta^{M_1} = B_r^{M_1} = 0 \end{cases} \tag{3}$$

whereas the corresponding forces have the following expression,

$$F_r^{M_1} = q v_\zeta^{M_1} B_\theta^{M_1}, \qquad F_\theta^{M_1} = 0, \qquad F_z^{M_1} = q v_r^{M_1} B_\theta^{M_1} \tag{4}$$

Note that, in this model, the longitudinal fields $E_z^{M_1}, B_z^{M_1}$ are identically zero.

In the same way, we denote by M_2 the second-order model corresponding to the asymptotic expansion of f up to the order 2, where order-3 terms have been neglected. This expansion is entirely determined from the first-order expansion $(F_r^{M_2}, F_\theta^{M_2}, F_z^{M_2})$ of the electromagnetic force. To compute it, the electromagnetic field associated with this model M_2 has to satisfy the following equations,

$$\begin{cases} E_r^{M_2} = cB_\theta^{M_2} = \dfrac{1}{\varepsilon_0 r} \displaystyle\int_0^r \rho^{M_2} s \, \mathrm{d}s \\ E_\theta^{M_2} = B_r^{M_2} = 0 \end{cases} \tag{5}$$

supplemented with, for the longitudinal fields:

$$\begin{cases} \dfrac{\partial E_z^{M_2}}{\partial r} = \dfrac{\partial B_\theta^{M_2}}{\partial t}, \\ E_z^{M_2}(r = R) = 0, \end{cases} \quad \text{and} \quad \begin{cases} \dfrac{\partial B_z^{M_2}}{\partial r} = \mu_0 J_\theta^{M_2} \\ \displaystyle\int_0^R B_z^{M_2} r \, \mathrm{d}r = 0 \end{cases} \tag{6}$$

Finally, the corresponding forces are expressed:

$$F_r^{M_2} = q\big(v_\theta^{M_2} B_z^{M_2} + v_\zeta^{M_2} B_\theta^{M_2}\big), \qquad F_\theta^{M_2} = -q v_r^{M_2} B_z^{M_2}, \qquad F_z^{M_2} = q\big(E_z^{M_2} + v_r^{M_2} B_\theta^{M_2}\big) \tag{7}$$

The models M_1 and M_2 are clearly simpler than the Maxwell one. Indeed, one replaces in each case the time-dependent Maxwell equations by a quasi-static model, where we have only to solve, for each ζ, two-dimensional transverse Poisson-like equations. This basically requires to compute integrals with respect to the variable r from the charge and current densities, see [29,9]. Moreover, as noted above, the bunch of particles is evolving slowly in a frame which moves along the optical axis at the speed of light. As a consequence, the computational domain can be defined as a simple rectangular domain in variables (r, ζ). The above equations will be easily solved by a finite-difference method written on a uniform rectangular mesh. A numerical analysis of the schemes and more details can be found in [29].

Concerning the Vlasov equation, it is classically solved by means of a particle method [1]. One approximates the function $rf(\mathbf{x}, \mathbf{p}, t)$ at any time t by a linear combination of delta distributions in the phase space $(\mathbf{x}, \mathbf{p})$, namely:

$$rf(\mathbf{x}, \mathbf{p}, t) = \sum_k w_k \delta\big(\mathbf{x} - \mathbf{x}_k(t)\big)\delta\big(\mathbf{p} - \mathbf{p}_k(t)\big) \tag{8}$$

where w_k denotes the constant weight of the particle k. Its position in the phase space $\mathbf{x}_k = (r, \zeta)_k$ and $\mathbf{p}_k = (p_r, p_\theta, p_z)_k$ is solution to a classical differential system, the solution of which being the characteristics of the Vlasov equations. It can be solved efficiently by a leap-frog scheme. The corresponding particle charge and current densities ρ and $\mathbf{J}$ are obtained by introducing the particle approximation (8) in Eqs. (1), which yields:

$$r\rho(\mathbf{x}, t) = q \sum_k w_k \delta\big(\mathbf{x} - \mathbf{x}_k(t)\big) \tag{9}$$

and

$$r\mathbf{J}(\mathbf{x}, t) = q \sum_k w_k \mathbf{v}_k(t) \delta\big(\mathbf{x} - \mathbf{x}_k(t)\big) \tag{10}$$

The Vlasov equation being discretized by a particle method and the M_1 or M_2 models being computed on a grid, the coupling between the fields and particles is obtained by gathering the contribution of the particles to the charge and current densities on the grid. In a reciprocal way, one has to interpolate the field values at the particles' positions.

In what follows, we will show on an example how the use of data mining techniques will help us to perform a kind of "sensitivity analysis". Our aim is to understand in which way model M_2 practically improves model M_1. As an example, we will use the particles' radial velocities $v_r^{M_i}$ $(i = 1, 2)$ as target variables in data mining analysis. An important remark is that the choice of these variables can not be automatic, and always depends on the human expertise that will decide what to show, verify or invalidate in the data.

Based on the methodology presented in the previous section, we will derive now a decision tree analysis for numerical computing applications. The first step consists in building the database. We consider the numerical results computed by

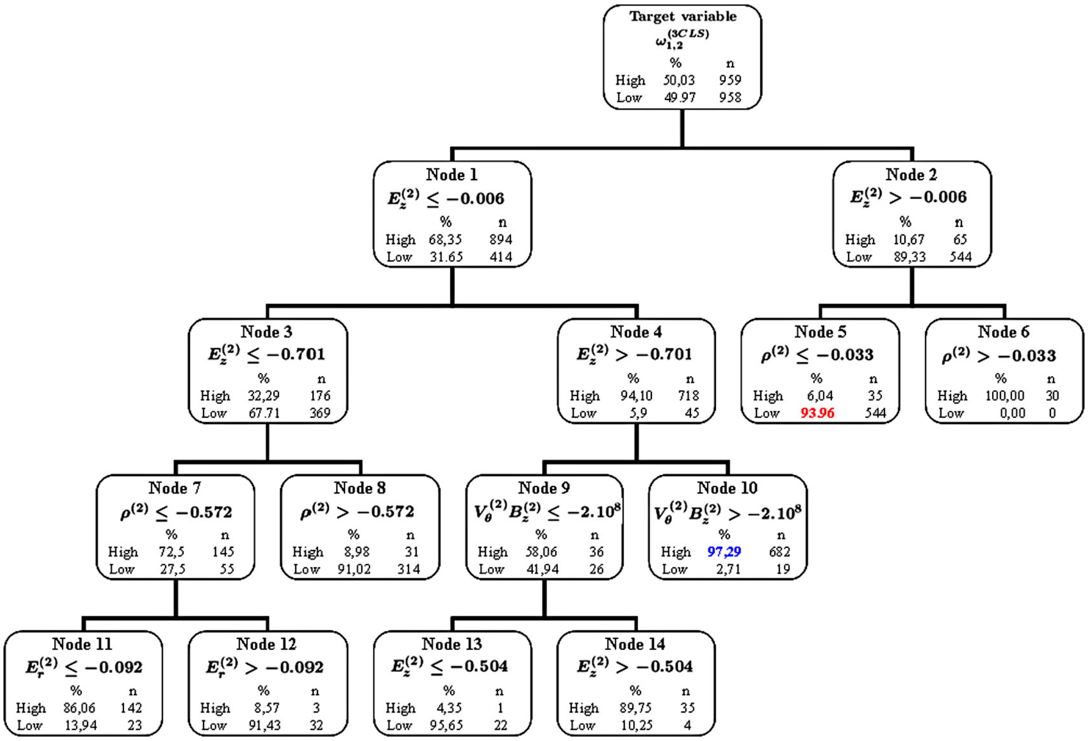

Fig. 1. (Color online.) Decision tree for v_r.

models M_1 and M_2 that we write together in the same database. Hence, at each time step and for each node of the space grid, we get a set of variables that are:

$$v_r^{M_i}, v_\theta^{M_i}, v_\zeta^{M_i}, E_r^{M_i}, E_z^{M_i}, B_z^{M_i}, \rho^{M_i}, J_\theta^{M_i}, F_r^{M_i}, F_\theta^{M_i}, F_z^{M_i} \quad (i = 1, 2) \tag{11}$$

Considering all the N_t time steps and the N_n space nodes, the database is composed by $N_t \times N_n$ rows and the 22 variables of (20) to be analyzed.

Our aim being to evaluate the contribution of the model M_2 compared to model M_1, we introduce a first variable

$$\omega_{1,2}(X) = \left| \frac{X^{M_1}}{X^{M_2}} \right| \tag{12}$$

that measures the weight of model M_1 in model M_2, regarding a given variable X. In practice, we are interested first in the case when $\omega_{1,2}$ is around 1: this corresponds to an equivalence of numerical results obtained between the two models M_1 and M_2 for the calculation of X. Then, when $\omega_{1,2}$ is either very small or very great compared to 1: this describes the situation when the numerical results between M_1 and M_2 are significantly different. From the variable $\omega_{1,2}$, we introduce our target variable denoted $\omega_{1,2}^{(LMH)}$, obtained by splitting the distribution of $\omega_{1,2}$ into three equal classes of individuals: Low, Medium and High. To identify which variables can explain a significant difference between X^{M_1} and X^{M_2}, we keep in our analysis the extreme groups defined by the "Low" class and the "High" class of the variable $\omega_{1,2}^{(LMH)}$.

Our aim is now to evaluate the differences between the distributions of the variables which should be meaningful between models M_1 and M_2. For this purpose, we computed the variable $\omega_{1,2}(v_r)$ and performed a decision tree on the target variable $\omega_{1,2}^{(LMH)}$, in order to identify the subgroups in the database which are homogenous to the categories "Low" and "High". We obtained the decision tree (Fig. 1). The precision of the decision tree, computed by the *risk estimate*, is equal to 6.05%, which means that 93.95% of data are correctly classified by the model of segmentation. Consequently, the quality level of the decision tree is very high. The first segmentation that appears shows that the most discriminating predictor variable is $E_z^{M_2}$, with a computed threshold equal to -0.006. This means that the "Low" $\omega_{1,2}(v_r)$ subgroup is mainly different from the "High" $\omega_{1,2}(v_r)$ subgroup, if one splits the whole involved population of the database up to the found threshold of $E_z^{(2)}$.

A first practical conclusion is that $v_r^{M_2}$ differs significantly from $v_r^{M_1}$ due to the variable $E_z^{M_2}$ in the asymptotic model M_2. This result was expected, since $E_z^{M_1} = 0$. On the other hand, in the classified list of predictors, the component $E_r^{M_2}$ appears as the second more significant predictor that explains the numerical difference between M_2 and M_1. This was less expected since the component $E_r^{M_1}$ is non-zero in model M_1, and can be explained because of the strong nonlinearity of the partial differential system. Finally, this classification also pointed out that $B_z^{M_2}$ does not appear as a predictor that could explain a significant difference between the two models. Even if B_z is equal to 0 in model M_1, namely $B_z^{M_1}=0$, $B_z^{M_2}$ does not bring a significant contribution to the model M_2, despite what was intuitively expected. Of course, it is always interesting to confirm such results with a formal approach, which analyzes the relation between $v_r^{M_1}$ and $v_r^{M_2}$. We refer the interested reader to [14] for more details.

5. Data mining to compare two numerical methods

In this second example, we will illustrate how data mining methods may help us to evaluate the error estimate in the case of the discretization error (see [21]). For introducing this error, let us consider a model of equations (E) that one wants to discretize with a numerical method in a given family of approximations, for instance the P_1 and P_2 finite-element method. The discretization error is defined as the error due to the difference of order between these methods.

At first glance, this definition appears quite trivial. Indeed, the Bramble–Hilbert theorem claims that, under certain conditions of regularity of the mesh and of the solution, the results obtained by finite elements P_2 (of order 2) will be more precise—in the sense of convergence order—than those computed by finite elements P_1 (of order 1). But this definition depends on the presence of the constant in the error estimates, generally unknown or difficult to estimate. In that case, as in the comparison of two models, we will show how data mining could help to investigate this problem.

For this purpose, we will introduce the third-order approximate model M_3 obtained by the procedure described above in Section 3. We will derive a particle approximation to deal with the Vlasov equation, whereas the model M_3 will be discretized successively by a P_1 and a P_2 finite-element conforming method. The coupling between these two approaches will be performed with an assignment and interpolation procedure, adapted to the P_1 and P_2 finite elements, respectively. Following [21], the system of equations M_3 is written (we drop the index ^{M_3} since there is no possible confusion):

$$\begin{cases} \dfrac{1}{r}\dfrac{\partial}{\partial r}(rE_r)=\dfrac{1}{\varepsilon_0}\rho(r,\zeta,t), \\ E_r(0,\zeta,t)=0, \end{cases} \qquad \begin{cases} \dfrac{\partial E_z}{\partial r}=\dfrac{1}{c}\dfrac{\partial E_r}{\partial t}, \\ E_z(R,\zeta,t)=0, \end{cases} \qquad \begin{cases} \dfrac{\partial B_z}{\partial r}=\mu_0 J_\theta(r,\zeta,t) \\ \displaystyle\int_0^R B_z r\,\mathrm{d}r=0 \end{cases}$$

$$\begin{cases} \dfrac{1}{r}\dfrac{\partial}{\partial r}(r\mathcal{E}_r)=\mu_0 c J_\zeta(r,\zeta,t)-\dfrac{1}{c}\dfrac{\partial E_z}{\partial t}, \\ \mathcal{E}_r(0,\zeta,t)=0, \end{cases} \qquad \begin{cases} \dfrac{1}{r}\dfrac{\partial}{\partial r}(r\mathcal{E}_\theta)=-\dfrac{\partial B_z}{\partial t} \\ \mathcal{E}_\theta(0,\zeta,t)=\mathcal{E}_\theta(R,\zeta,t)=0 \end{cases} \tag{13}$$

where J_ζ is defined by $J_\zeta=\rho c-J_z=q\int v_\zeta f\,\mathrm{d}\mathbf{v}$.

In this model appears the additional unknowns $\mathcal{E}_r=E_r-cB_\theta$ and $\mathcal{E}_\theta=E_\theta+cB_r$. The expression of the second-order expansion of the Lorentz force $\mathbf{F}=(F_r,F_\theta,F_z)$ is given by

$$\begin{cases} F_r=q(\mathcal{E}_r+v_\theta B_z+v_\zeta E_r/c) \\ F_\theta=q(\mathcal{E}_\theta-v_r B_z) \\ F_z=q(E_z+v_r E_r/c) \end{cases} \tag{14}$$

and entirely determines the asymptotic expansion of f up to the order 3. We have now to build P_1 and P_2 finite-element approximations to explore the corresponding results by data mining techniques. In what follows, we briefly present the derivation of the variational formulation [30]. This is quite standard, except for component B_z. Moreover, as the regularity of the fields is not an issue for our study, we will assume that they are smooth enough. For the sake of simplicity, we will denote by V the space of the fields and of the test functions, regardless of the boundary conditions they satisfy.

We readily obtain, for the electric component E_r, *find $E_r\in V$ such that*

$$\int_0^Z\int_0^R \frac{\partial}{\partial r}(rE_r)v\,\mathrm{d}r\,\mathrm{d}\zeta=\frac{1}{\varepsilon_0}\int_0^Z\int_0^R \rho v r\,\mathrm{d}r\,\mathrm{d}\zeta, \quad \forall v\in V \tag{15}$$

together with the boundary condition $E_r(0,\zeta,t)=0$. Similarly, for the E_z component, *find $E_z\in V$ such that*

$$\int_0^Z\int_0^R \frac{\partial E_z}{\partial r}v\,\mathrm{d}r\,\mathrm{d}\zeta=\frac{1}{c}\int_0^Z\int_0^R \frac{\partial E_r}{\partial t}v\,\mathrm{d}r\,\mathrm{d}\zeta, \quad \forall v\in V \tag{16}$$

together with the perfect conductor boundary condition $E_z(R,\zeta,t)=0$.

Remark 5.1. One can choose to handle the boundary condition by using an integration by part formula regarding r. We get:

$$-\int_0^Z\int_0^R E_z\frac{\partial v}{\partial r}\,\mathrm{d}r\,\mathrm{d}\zeta-\int_0^Z E_z v(r=0,\zeta,t)\,\mathrm{d}\zeta=\frac{1}{c}\int_0^Z\int_0^R\frac{\partial E_r}{\partial t}v\,\mathrm{d}r\,\mathrm{d}\zeta,\quad\forall v\in V$$

The difficulty to compute the magnetic component B_z is related to its boundary condition, which has an integral form. To overcome this difficulty, we introduce the variable $\mathcal{B}_z(r,\zeta,t)$ defined by

$$\mathcal{B}_z(r,\zeta,t):=\int_0^r B_z(s,\zeta,t)s\,\mathrm{d}s \tag{17}$$

so that third equation of (13) is written as

$$\frac{\partial^2\mathcal{B}_z}{\partial r^2}-B_z=\mu_0 rJ_\theta$$

whereas the integral boundary condition gives $\int_0^R B_z(s,\zeta,t)r\,\mathrm{d}r=\mathcal{B}_z(R,\zeta,t)=0$. Hence, one equivalently replaces the third equation of (13) by

$$\begin{cases} r\dfrac{\partial^2\mathcal{B}_z}{\partial r^2}-\dfrac{\partial\mathcal{B}_z}{\partial r}=\mu_0 r^2 J_\theta\\ \mathcal{B}_z(R,\zeta,t)=0\\ \dfrac{\partial\mathcal{B}_z}{\partial r}(0,\zeta,t)=0\end{cases} \tag{18}$$

A variational formulation for $\mathcal{B}_z(r,\zeta,t)$ is given by: *find $\mathcal{B}_z\in V$ such that*:

$$-\int_0^Z\int_0^R\frac{\partial\mathcal{B}_z}{\partial r}\frac{\partial v}{\partial r}r\,\mathrm{d}r\,\mathrm{d}\zeta+R\int_0^Z\frac{\partial\mathcal{B}_z}{\partial r}v(R,\zeta,t)\,\mathrm{d}\zeta-\int_0^Z\int_0^R\frac{\partial\mathcal{B}_z}{\partial r}v\,\mathrm{d}r\,\mathrm{d}\zeta=\mu_0\int_0^Z\int_0^R r^2J_\theta v\,\mathrm{d}r\,\mathrm{d}\zeta,\quad\forall v\in V \tag{19}$$

The advantage of this formulation is that only $\mathcal{U}:=\frac{\partial\mathcal{B}_z}{\partial r}$ is involved and can be chosen as a new unknown. Moreover, the boundary condition $\frac{\partial\mathcal{B}_z}{\partial r}(0,\zeta,t)=0$ is handled (in a weak way) through the integration by parts.

Remark 5.2. Solving a variational formulation in $\mathcal{U}$ leaves an indetermination in the computation of $\mathcal{B}_z$. Indeed, $\mathcal{B}_z$ is *a priori* determined up to an additive constant. Using the boundary condition $\mathcal{B}_z(R,\zeta,t)=0$ allows us to uniquely determine $\mathcal{B}_z$, for instance by choosing $\mathcal{B}_z(r,\zeta,t)=\int_R^r\mathcal{U}(r,\zeta,t)\,\mathrm{d}s$.

The variational formulations for unknowns $\mathcal{E}_r$ and $\mathcal{E}_\theta$ are straightforward and similar to the one of E_r. From these variational formulations, one derives the finite-element conforming P_1 and P_2 approximations in an efficient way by using the FreeFem++ package [31]. The Vlasov equation is solved as in Section 4 by a particle method. The coupling between the P_1 finite-element approximation and the particle method is performed following the classical assignment and interpolation procedures [1,32]. Nevertheless, the coupling is less standard for the P_2 finite element. We will briefly describe it. Let us denote by $\{\mathbf{a}_i=(r_i,\zeta_i)\}$ the set of mesh vertices where the mesh values ρ^M and $\mathbf{J}^M$ are needed, namely the vertices and the middle of the edges of the triangular mesh. Let χ_i be the finite element basis functions and let $\mathbb{M}$ be the mass matrix with entries $\mathbb{M}_{i,j}=\int_\Omega\chi_i\chi_j r\,\mathrm{d}r\,\mathrm{d}\zeta$. The values of the charge and the current densities at the nodes of the mesh are defined by

$$\sum_j\mathbb{M}_{i,j}\rho^M(\mathbf{a}_j,t)=\int_\Omega\rho(\mathbf{x},t)\chi_i r\,\mathrm{d}r\,\mathrm{d}\zeta=q\sum_{k\in\mathcal{K}_{\mathbf{a}_i}}w_k\chi_i\big(\mathbf{x}_k(t)\big)$$

and

$$\sum_j\mathbb{M}_{i,j}\mathbf{J}^M(\mathbf{a}_i,t)=\int_\Omega\mathbf{J}(\mathbf{x},t)\chi_i r\,\mathrm{d}r\,\mathrm{d}\zeta=q\sum_{k\in\mathcal{K}_{\mathbf{a}_i}}w_k\mathbf{v}_k(t)\chi_i\big(\mathbf{x}_k(t)\big)$$

where $\mathcal{K}_{\mathbf{a}_i}$ denotes the set of particles located in the element that has $\mathbf{a}_i$ as a vertex. Since $\sum_i\chi_i\equiv 1$, this procedure preserves the total charge and the total current of particles.

Our aim is now to show on this example how the use of data mining techniques will help us to compare the P_1 and P_2 finite-element method. As in the previous example, we collect in a database all the numerical results computed by the two methods. Since they have not the same degrees of freedom, we chose to keep in the database the common data, that is the quantities computed at the vertices of the mesh. This excludes, for the P_2 finite element, the quantities computed at the middle of the edges of the mesh. Hence, our database is composed by

Table 1
Respective proportions of the "P_1 *vs.* P_2" categories.

	Count	Percent
Different Order	6522	86.3
Same Order	1035	13.7

$$v_r^i, v_\theta^i, v_\zeta^i, E_r^i, E_z^i, B_z^i, \mathcal{E}_r^i, \mathcal{E}_\theta^i, \rho^i, J_r^i, J_\theta^i, J_\zeta^i, F_r^i, F_\theta^i, F_z^i \quad (i = 1, 2) \tag{20}$$

computed at each time step and for each node of the mesh, the exponent i specifying here the P_1 or P_2 finite-element approximations.

Our objective is to appreciate the difference of accuracy between P_1 and P_2 solutions. To this end, we have to determine if there exist rows in the database such that P_1 method would provide a better approximation than the P_2 finite-element one or, at least, an equivalent one. In the absence of the exact solution, we cannot determine what "better approximation" means. So we try to identify subgroups in the database such that the numerical approximations computed by P_1 and P_2 finite elements are of "*the same order*".

For this example, let us restrict ourselves to the radial component of the electrical field E_r. To identify the approximations of "*the same order*", we will only keep in the database the rows such that neither E_r^1 nor E_r^2 are too small. This is to eliminate two possible situations:

- the case where E_r^1 is small with respect to E_r^2 or reciprocally. This case means that E_r^1 and E_r^2 are not of "*the same order*";
- the case where E_r^1 and E_r^2 are both small. Again, in absence of the exact solution, this corresponds to the situation where we cannot determine if E_r^1 and E_r^2 are or not of "*the same order*".

In terms of numerical values, each component E_r^1 and E_r^2 is assumed to be "small" at a given time step t_n and for a given node (r_j, ζ_k), if its value is smaller than 5% of the maximum of all the values of E_r^1 and E_r^2 respectively. The threshold of 5% is arbitrary yet reasonable, and a sensitivity study about this choice is on progress. Applying this rule allows us to extract from the database the rows one has to explore.

At this time, we have to define the notion of "same numerical order". For this purpose, we introduce a qualitative binomial variable "P_1 *vs.* P_2" as follows:

$$P_1 \textit{ vs. } P_2 \equiv \left| \begin{array}{ll} \textit{Same Order}, & \text{if } \left|E_r^2 - E_r^1\right| \leq \alpha \\ \textit{Different Order}, & \text{if not} \end{array} \right. \tag{21}$$

where α denotes a given threshold. In these conditions, the value "Same Order" of "P_1 *vs.* P_2" will allow us to detect and characterize situations where using P_2 finite-element method does not improve significantly the computation of E_r^1. Now, taking in this example the threshold $\alpha = 0.65$, which corresponds to 5% of the maximum of the absolute difference between E_r^1 and E_r^2, we process data mining techniques to qualify the two different categories of the target variable (*i.e.* the variable to be explained) "P_1 *vs.* P_2". Results in Table 1 show that the dataset contains a non-negligible quantity of elements (almost 14%) such that P_1 *vs.* P_2 is of the same order.

Now, to qualify these 14% of "Same Order" elements, we could process data mining techniques, guided by what we are aiming to verify. In this example, one can use, among others, a decision tree, as in Section 4. The first segmentation that appears on the decision tree highlights the most discriminated predictor variable. One can observe that the time is detected as this predictor, with a corresponding computed optimal threshold equal to the 42nd time step over a hundred computed. This means that when the time is smaller than the 42nd time step, the corresponding blue node in the decision tree is very homogeneous regarding the variable "Same Order". More precisely, almost 80% of the elements of this node have this value for the target variable "P_1 *vs.* P_2". We can then conclude that for time steps smaller than the 42nd one, the implementation of P_2 finite elements was over qualified and then, the cost of the computation could not be justified anymore. At the opposite, when one considers time steps that are greater than the 42nd, the corresponding cluster, which is the other node in the decision tree (see Fig. 2), is constituted by more than 98% of rows that correspond to the value "Different Order" of the target variable "P_1 *vs.* P_2". Obviously, this kind of investigation needs to be deepened and generalized. This is our current and future work. Nevertheless, this illustrates to our opinion how data mining can contribute to solving efficiently Vlasov–Maxwell equations.

6. Conclusion

In this paper, we have proposed to use data mining methods to evaluate numerical solutions of Vlasov–Maxwell equations. After a brief presentation of the data mining principles, we gave two examples to illustrate how this idea can be processed.

In a first example, we focused our study to the case of asymptotic paraxial models. Our aim was to determine the role of the different powers in the asymptotic expansion, considering the two first models M_1 and M_2. The results we have

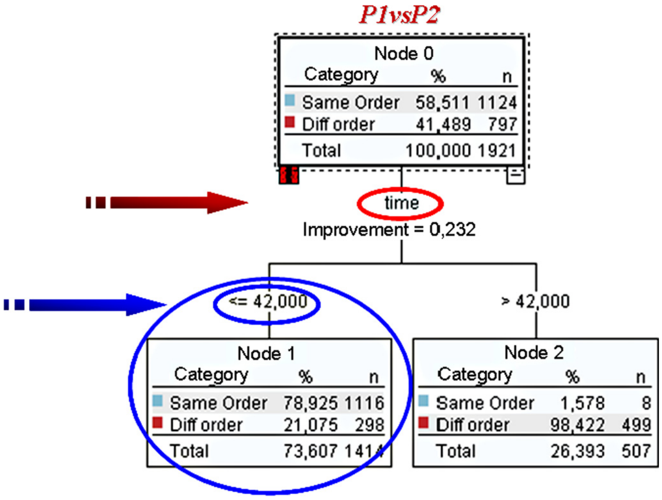

Fig. 2. (Color online.) Decision tree.

obtained suggest that data mining techniques can be applied to the analysis of Vlasov–Maxwell approximations as it is applied in a lot of other domains.

In a second part, we gave an example of comparisons between two numerical methods that approximate the Vlasov–Maxwell equations. We described the mathematical model and the variational formulation. Then, we constructed a database made of the numerical results obtained by a P_1 and a P_2 finite-element method, coupled with a particle method. Based on Decision Tree techniques, we identified and characterized the elements such that the evaluation between the two methods gave similar numerical results.

Obviously, the conclusion we got on these two error evaluations is yet partial and needs to be confirmed, for instance by investigating other variables. Moreover, we have to keep in mind that conclusions are also to be guided by the human expertise, that has to be involved in the data exploration. It also basically depends on the purpose of the performed analysis. However, this illustrates how data mining can actually help to evaluate several kinds of errors for Vlasov–Maxwell equations.

References

[1] C.K. Birdsall, A.B. Langdon, Plasmas Physics via Computer Simulation, MacGraw-Hill, New York, 1985.

[2] P. Degond, P.-A. Raviart, On the paraxial approximation of the stationary Vlasov–Maxwell, Math. Models Methods Appl. Sci. 3 (4) (1993) 513–562.

[3] E. Frénod, E. Sonnendrücker, The finite Larmor radius approximation, SIAM J. Math. Anal. 32 (6) (2001) 1227–1247.

[4] G. Laval, S. Mas-Gallic, P.-A. Raviart, Paraxial approximation of ultrarelativistic intense beams, Numer. Math. 69 (1) (1994) 33–60.

[5] M.A. Mostrom, D.I. Mitrovich, D.I.R. Welch, The ARCTIC charged particle beam propagation code, J. Comput. Phys. 128 (2) (1996) 489–497.

[6] P.A. Raviart, E. Sonnendrücker, A hierarchy of approximate models for the Maxwell equations, Numer. Math. 73 (3) (1996) 329–372.

[7] S. Slinker, G. Joyce, J. Krall, R.F. Hubbard, ELBA – a three dimensional particle simulation code for high current beams, in: Proc. of the 14th Int. Conf Numer. Simul. Plasmas, Annapolis, MD, USA, 1991.

[8] F. Assous, P. Degond, J. Segré, A particle-tracking method for 3D electromagnetic PIC codes on unstructured meshes, Comput. Phys. Commun. 72 (6) (1992) 105–114.

[9] F. Assous, F. Tsipis, Numerical paraxial approximation for highly relativistic beams, Comput. Phys. Commun. 180 (2009) 1086–1097.

[10] N. Besse, G. Latu, A. Ghizzo, E. Sonnendrücker, P. Bertrand, A wavelet-MRA-based adaptive semi-Lagrangian method for the relativistic Vlasov–Maxwell system, J. Comput. Phys. 227 (16) (2008) 7889–7916.

[11] N. Crouseilles, M. Mehrenberger, E. Sonnendrücker, Conservative semi-Lagrangian schemes for the Vlasov equation, J. Comput. Phys. 229 (2010) 1927–1953.

[12] F. Filbet, Convergence of a finite volume scheme for the one dimensional Vlasov–Poisson system, SIAM J. Numer. Anal. 39 (4) (2001) 1146–1169.

[13] Y. Saad, From data mining to nanotechnology and back: the new problems of numerical linear algebra, in: Computer Science & Eng. Colloquium, University of Minnesota, November 2009.

[14] F. Assous, J. Chaskalovic, Data mining techniques for scientific computing: application to asymptotic paraxial approximations to model ultra-relativistic particles, J. Comput. Phys. 230 (2011) 4811–4827.

[15] O. Kulski, J. Chaskalovic, et al., Explicative factors for prognostics IIU: exploration on 2089 cycles done with statistical and data mining tools, in: 9th Meeting of the French Federation of the Reproduction Studies, Palais des Congrès, Paris, 2004.

[16] X.L. Nguyên, J. Chaskalovic, et al., Residual subjective daytime sleepiness under CPAP treatment in initially somnolent apnea patients: a pilot study using data mining methods, Sleep Med. 9 (5) (2007) 511–516.

[17] X.L. Nguyên, J. Chaskalovic, et al., Insomnia symptoms and CPAP compliance in OSAS patients: a descriptive study using data mining methods, Sleep Med. 11 (8) (2010) 777–784.

[18] R. Lefébure, G. Venturi, Data Mining – Gestion de la Relation Client, Eyrolles, Paris, 2001.

[19] J. Chaskalovic, A new approach in media/marketing databases explorations for application in e-business, in: National Congress of IREP, Paris, 1999.

[20] J. Chaskalovic, A. Vanheuverzwyn, Innovation in estimations: a reliable approach for radio audience indicators, in: Proc. Esomar, WM^3 2007, Dublin, 3–6 June 2007.

[21] F. Assous, J. Chaskalovic, Error estimate evaluation in numerical approximations of partial differential equations: a pilot study using data mining methods, C. R. Mecanique 341 (2013) 304–313.

[22] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edn., Springer Series in Statistics, 2009.
[23] T. Kohonen, Self-Organizing Maps, 3rd edn., Springer Series in Inf. Sci., vol. 30, Springer, Berlin, 2001.
[24] U. Seiffert, L. Jain, Self-Organizing Neural Networks: Recent Advances and Applications, Physica Verlag, Heidelberg, Germany, 2002.
[25] L. Rokach, O. Maimon, Data Mining with Decision Trees: Theory and Applications, World Scientific Publishing Company, 2001.
[26] A.J. Berkhout, Seismic Migration, Elsevier, Amsterdam, Oxford, New York, Tokyo, 1984.
[27] J.F. Claerbout, Imaging the Earth's Interior, Blackwell Scientific Publications, Oxford, 1985.
[28] F. Assous, J. Chaskalovic, A new paraxial asymptotic model for the relativistic Vlasov–Maxwell equations, C. R. Mecanique 340 (2012) 706–714.
[29] F. Assous, F. Tsipis, A PIC method for solving a paraxial model of highly relativistic beams, J. Comput. Appl. Math 227 (1) (2009) 136–146.
[30] J. Chaskalovic, Mathematical and Numerical Methods for Partial Differential Equations, Springer Verlag, 2014.
[31] F. Hecht, FreeFem++, Numerical mathematics and scientific computation 3.7, Laboratoire Jacques-Louis-Lions, Université Pierre-et-Marie-Curie, Paris, 2010, http://www.freefem.org/ff++/.
[32] R.W. Hockney, J.W. Eastwood, Computer Simulation Using Particles, Adam Hilger imprint by IOP Publishing Ltd., 1988.