A Parareal algorithm for a highly oscillating Vlasov-Poisson system with reduced models for the coarse solving

Laura Grigori, Sever A Hirstoaga, Julien Salomon

A Parareal algorithm for a highly oscillating Vlasov-Poisson system with reduced models for the coarse solving

Laura Grigori * Sever A. Hirstoaga * Julien Salomon †

December 2022

Abstract

In this paper, we introduce a new strategy for solving highly oscillatory two-dimensional Vlasov-Poisson systems by means of a specific version of the parareal algorithm. The novelty consists in using reduced models, obtained from the two-scale convergence theory, for the coarse solving. The reduced models are useful to approximate the original Vlasov-Poisson model at a low computational cost since they are free of high oscillations. Both models are numerically solved in a particle-in-cell framework. We illustrate this strategy with numerical experiments based on long time simulations of a charged beam in a focusing channel and under the influence of a rapidly oscillating external electric field. On the basis of computing times, we provide an analysis of the efficiency of the parareal algorithm in terms of speedup.

Keywords: Vlasov-Poisson system, two-scale convergence, multi-scale models, particle-in-cell method, parareal algorithm, parareal speedup

1 Introduction

The present contribution focuses on the construction of a parallel-in-time numerical scheme for an accurate and efficient solving in long term of a kinetic equation which involves several time scales. The kinetic description under consideration is a Vlasov-Poisson system which models the dynamics of a charged particle beam, focused by an external electric field and under the action of the self-consistent electric force. Specifically, the system reads

$$\frac{\partial f_\varepsilon}{\partial t} + \frac{1}{\varepsilon} v \frac{\partial f_\varepsilon}{\partial r} + \Big(E_\varepsilon + \Xi_\varepsilon \Big) \frac{\partial f_\varepsilon}{\partial v} = 0, \quad f_\varepsilon(t=0,r,v) = f_0(r,v), \tag{1}$$

where $0 < \varepsilon \ll 1$ is a parameter, $f_\varepsilon = f_\varepsilon(t,r,v) > 0$ is the distribution function of a particle species, $t \in [0,T)$ is the time, $r \in [r_{\min}, r_{\max}]$ is the position and $v \in \mathbb{R}$ the velocity, while f_0 is a given initial condition. The functions $\Xi_\varepsilon = \Xi_\varepsilon(t,r) \in \mathbb{R}$ and $E_\varepsilon = E_\varepsilon(t,r) \in \mathbb{R}$ play the

*Inria, Project-Team ALPINES, Sorbonne Université and Université de Paris, CNRS, Laboratoire Jacques-Louis Lions (LJLL), 75589 Paris Cedex 12. E-mails: laura.grigori@inria.fr, sever.hirstoaga@inria.fr

†Inria, Project-Team ANGE, Sorbonne Université and Université de Paris, CNRS, Laboratoire Jacques-Louis Lions (LJLL), 75589 Paris Cedex 12. E-mail: julien.salomon@inria.fr

role of external and self-consistent electric forces, respectively. More precisely, we consider the term Ξ_ε under the form

$$\Xi_\varepsilon(t,r) = -\frac{r}{\varepsilon} + rH\Big(\frac{t}{\varepsilon}\Big), \tag{2}$$

where H is a given 2π-periodic function, while the self-consistent electric field is solution to the following Poisson equation

$$\frac{1}{r}\frac{\partial(rE_\varepsilon)}{\partial r} = \rho_\varepsilon, \text{ where } \quad \rho_\varepsilon(t,r) = \int_{\mathbb{R}} f_\varepsilon(t,r,v)\,dv. \tag{3}$$

This model is derived as a paraxial approximation of a full Vlasov-Maxwell system and under several assumptions on the physical properties of the beam (see [8, 10] and the references therein). An example of such assumption is that the beam is long and thin, meaning that the longitudinal self-consistent forces are neglected and that the transverse dimensions of the beam are small compared to the characteristic longitudinal dimension. Besides, the small parameter ε denotes the ratio between these two characteristic lengths.

The coupled Vlasov-Poisson equations (1)-(2)-(3) have been extensively studied theoretically as well as numerically (see for example [8, 10, 21, 5, 6, 9]). The difficulty arising in the numerical solving of this system is that the solution displays high oscillations in time when the parameter ε is small. Thus, any standard numerical method needs to resolve these oscillations by using a time step which is much smaller than ε, which corresponds to the order of the period of oscillation. Consequently, this approach has a high computational cost, which is prohibitive in applications demanding long run times. In addition, applying some of the conventional numerical methods (which are not symplectic for example) leads to cumulative large errors in the solution.

Several works deal with these issues. In [5, 6], the authors design finite difference and semi-Lagrangian schemes whose level of accuracy is uniform with respect to the oscillations frequencies, for well-chosen initial conditions. This approach gives excellent results for long time simulations where the particle beam is shorter than the case considered in the present contribution. Another method, based on an exponential time integrator of the characteristics of the Vlasov equation, is proposed in [9] (see also the references therein). However, this last paper tackles only the case of equations (1)-(2)-(3) with $H = 0$ and with a particle beam which is short, as in references [5, 6]. The case of a shorter particle beam is simpler than the one considered in the present work since its behaviour is much easier to capture by a reduced (or averaged) model that we introduce in Section 2.

This paper aims at building a numerical frame that provide an accurate long time solution of (1)-(2)-(3), without the computational burden inherent to standard schemes. We achieve this goal in two steps. First, we infer two reduced models (corresponding to two choices for the function H) from the limit model obtained in [10] *via* the two-scale convergence theory. The benefits of the reduced models are: *(i)* they approximate over small time intervals the initial Vlasov-Poisson model in the limit $\varepsilon \to 0$; *(ii)* they are significantly less computationally expensive than the full model; see [22] for a general theory of reduced (or averaging) models for dynamical systems. Their drawback is that in long time the approximation becomes inaccurate. To circumvent this issue, we propose in a second step to use the parareal algorithm [18]. This framework allows to correct the reduced models by applying the fine (or full) models only in short time windows and in parallel. The parareal

algorithm has been intensively analyzed, see, *e.g.*, [2, 12, 23, 11] and applied in various contexts, as fluid and structure problems [7], the Navier–Stokes equations [13], or reservoir simulation [14]. Several variants of the method have also been proposed [14, 7], including its adaptation to matrix function computations [4]. The algorithm exploits very efficiently parallel computing over a large number of processors to solve problems subject to time constrains. This approach is detailed in the next section.

The strategy of using reduced models in a parareal algorithm goes back at least to [19] (see also [16, 1]) and was successfully applied in our previous paper [15] where we solved differential equations which are similar to the characteristics of the equation (1). Specifically, there is only an external electric term in the equations in [15] and no self-consistent field is present. The new framework of the Vlasov-Poisson system cannot be considered as a consequence of the method in [15] since the self-consistent electric term is not an external force but changes in time according to the unknown distribution function. Thus, the limit models considered in the previous reference cannot be used in the new configuration. Instead, one needs to use a specific reduced model which takes into account the correct two-scale limit electric field. This limit is obtained by following the approach derived in [10].

Finally, we briefly describe the numerical scheme that we choose for solving the system (1)-(2)-(3) and the reduced models, namely the particle-in-cell method [3, 17]. This approach is based on the approximation of the unknown distribution function by a sum of Dirac masses, or in other words, by a collection of numerical particles in the phase space (r, v). Afterwards, we have to advance in time these particles by solving the differential equations given by the characteristics of the Vlasov equation. Here, we need to solve the Poisson equation at each time step to take into account the interaction between particles, *i.e.*, the self-consistent electric field.

The remainder of the paper is organized as follows. In section 2, we describe the problem under consideration and the numerical strategy in the parareal framework. In section 3, we first discuss the two-scale limit model obtained in [10] and we propose a new reduced model to be used in the case of $H \neq 0$. Then, we present the particle-in-cell method and study the validity of approximation of the reduced models in long time. In section 4, we analyze the results obtained with our strategy. We demonstrate that the parareal algorithm only requires few iterations to converge and discuss the efficiency of the method according to the values of the parameters involved. More precisely, we obtain a low computational cost of the method, in large final times and for any parameter ε, while keeping the accuracy of the fine solver.

2 The problem of interest and the parareal algorithm

We first provide an illustrative description of problem (1)-(2)-(3). The unknown of the Vlasov-Poisson system is the pair $(f_\varepsilon, E_\varepsilon)$, while Ξ_ε is a given function. In the numerical experiments, we consider two cases: first, $H \equiv 0$ (called from now on Case I), and second (Case II) for

$$H(\tau) = \cos^2(\tau). \tag{4}$$

We explain in the next section why these two cases are well representative for the general behaviour of the solution. Note that similar tests could be performed with other 2π-periodic functions. The choice of H impacts significantly on the dynamics of the particles. For example, the case of the function $H(\tau) = \cos(\tau)$ leads to a behaviour of the solution similar to the case $H \equiv 0$.

The problem models the dynamics of a beam of particles that move in the phase space (r, v) along the characteristic curves of the Vlasov equation (1). This dynamics entails a fast rotation around the origin (at the time scale ε), coupled to slower motions which are due to the self-consistent electric field E_ε and to the external force term H. The behaviour of the solutions to (1)-(2)-(3) is illustrated in Figs. 1 and 2 for both Case I and Case II, where one can see filaments of particles that develop differently depending on the external force H. Note that the solutions represented in these figures are obtained with the initial condition in (20). However, in [6, 9] a simpler case is considered where the space interval $[r_{\min}, r_{\max}]$ is smaller, corresponding to a shorter particle beam. With such an initial data, the particle beam evolves differently than those in Figs. 1 & 2, by rotating around the origin and spiraling from the origin in long times.

In this context, the role of a reduced model is to approximate the slow motions without resolving the fastest rotation. The computational cost of such a model is therefore low. We first illustrate these ideas on an example in a general context. Given $\alpha > 0$, consider the problem

$$\frac{\mathrm{d}u_\varepsilon}{\mathrm{d}t} = F\Big(\frac{t}{\varepsilon}, u_\varepsilon\Big) \quad \text{in } (0,T),\ u_\varepsilon(0) = \mathrm{u}_0, \tag{5}$$

where ε is a small parameter as above, F is a regular enough function which is α-periodic with respect to its first variable. Then, the averaged (or reduced) model associated to (5) is given by (see [22])

$$\frac{\mathrm{d}v}{\mathrm{d}t} = G(v) \quad \text{in } (0,T),\ v(0) = \mathrm{u}_0, \quad \text{where } G(v) = \frac{1}{\alpha}\int_0^\alpha F(t,v)dt. \tag{6}$$

In this setting, it is well-known [22] that one can derive the approximation $\|u_\varepsilon(t) - v(t)\| \leq C\varepsilon$ for $t \in (0,T)$, where the constant C and the final time T are independent of ε. The important point to be remarked is that equation (6) can be solved with large time steps with respect to ε. Back to the problem (1)-(2)-(3), we make use in this paper of the reduced model developed in [10] to approximate the initial stiff Vlasov-Poisson system. However, for a long time interval (for instance a thousand of fast rotations), such an averaged model turns out to be not accurate enough, even for small values of ε. A solution to perform accurate long time simulations of stiff systems, with low computational cost, can be the use of the reduced model in a parareal framework, as described below.

In order to accelerate the solving of our problem, we will make use of the parareal algorithm. Let us briefly recall the main features of this approach in the framework of the time dependent problem (5) and its limit (6). The time interval $[0,T]$ is decomposed into N uniform time intervals $[T_n, T_{n+1}]$, for $n \in \{0, \ldots, N-1\}$. Let $\mathcal{F}(T_{n+1}, T_n, Y_n)$ denote the fine solver, which is assumed to provide a very accurate approximation of the solution of (5) at time T_{n+1} with the initial condition Y_n given at time T_n. In the same way, let $\mathcal{G}(T_{n+1}, T_n, Y_n)$ denote the coarse solver, which is assumed to provide a coarse

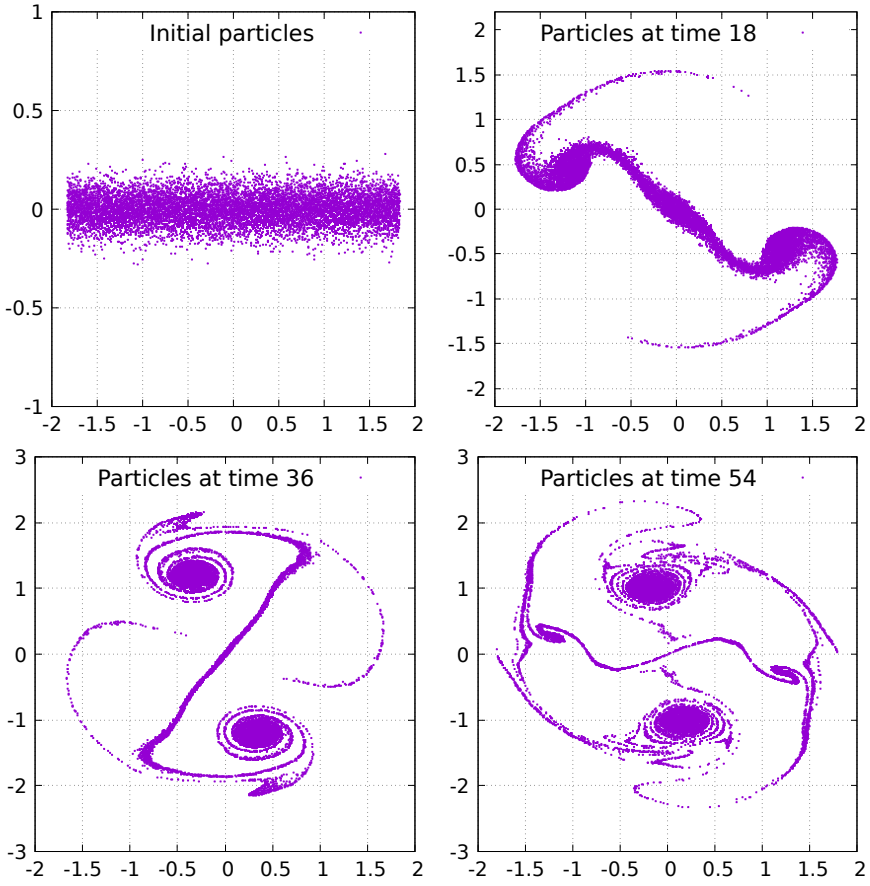

Figure 1: The particle solution of the Vlasov-Poisson Case I with $\varepsilon = 0.01$ at different times $T \in \{0, 18, 36, 54\}$.

approximation of the solution of (5) at time T_{n+1} also with the condition Y_n given at time T_n. In this framework, the use of $\mathcal{F}$ for solving equation (5) requires a computational effort, since the time step is constrained by the small parameter ε. Therefore, the coarse solver must be chosen in such a way that its cost is much lower than the one of the fine solver. A standard strategy to achieve this requirement consists in using as a coarse solver either the approximation scheme considered in the fine solver but with a larger time step or a different approximation scheme with lower accuracy [12]. Yet, another approach is to use a different model from the initial one, as long as it provides a reasonable coarse and fast approximation of the solution [19, 15].

In this paper, we follow this last approach and use as a coarse solver a reduced model associated with the original problem. In this way, the coarse solver $\mathcal{G}(T_{n+1}, T_n, Y_n)$ is always assigned to the numerical solution of the reduced model (6) and the fine solver $\mathcal{F}(T_{n+1}, T_n, Y_n)$ is always assigned to the approximated solution of the original problem (5). In addition, we let the coarse propagator perform a single time step per time interval $[T_n, T_{n+1}]$. The parareal algorithm then consists in computing iteratively a sequence $(Y_n^k)_{k,n}$ of approxima-

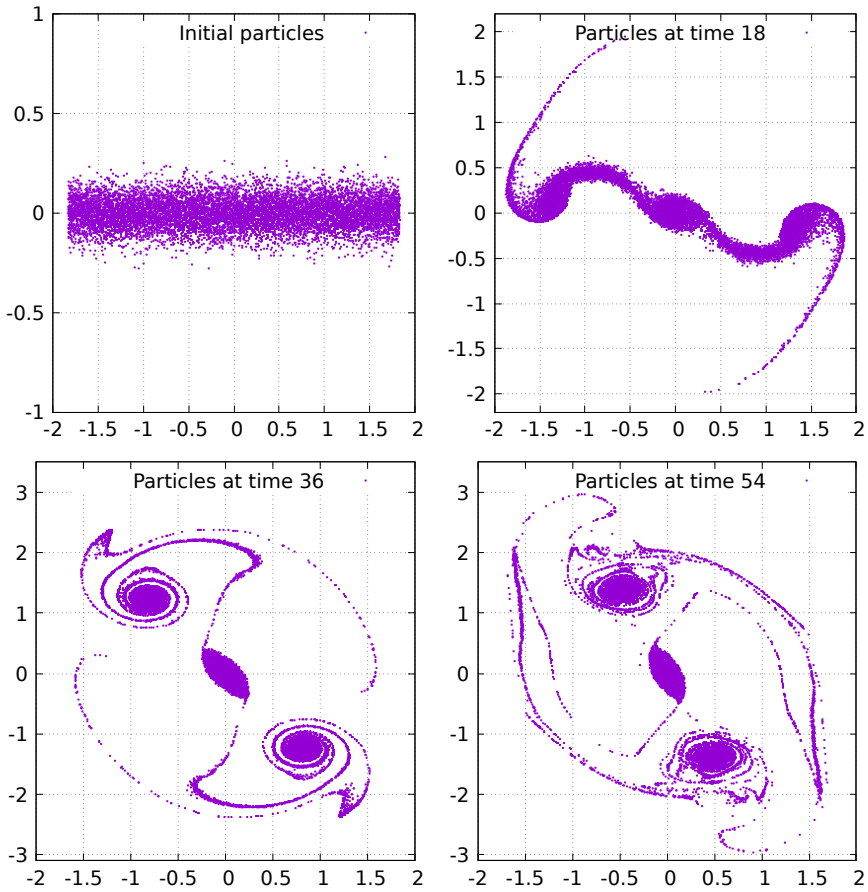

Figure 2: The particle solution of the Vlasov-Poisson Case II (with the H function in (4)) and with $\varepsilon = 0.01$ at different times $T \in \{0, 18, 36, 54\}$.

tions of $u_\varepsilon(T_n)$ for $n \in \{0, \dots, N\}$ that will serve as initial condition on $[T_n, T_{n+1}]$, where a fine solving will be carried out in parallel. More precisely, it proceeds as follows. In a first step, the initial approximation Y_n^0 at coarse time points $0 = T_0 < T_1 < \cdots < T_N = T$ is computed sequentially using the coarse solver, namely

$$Y_{n+1}^0 = \mathcal{G}(T_{n+1}, T_n, Y_n^0), \;\; Y_0^0 = \mathrm{u}_0. \tag{7}$$

Then, for $k = 0, 1, \dots$, starting with $Y_0^{k+1} = \mathrm{u}_0$, the sequence $(Y_n^k)_{k,n}$ is updated by

$$Y_{n+1}^{k+1} = \mathcal{G}(T_{n+1}, T_n, Y_n^{k+1}) + \mathcal{F}(T_{n+1}, T_n, Y_n^k) - \mathcal{G}(T_{n+1}, T_n, Y_n^k), \;\; \forall n \in \{0, \dots, N-1\}. \tag{8}$$

In this iteration, the terms $\mathcal{F}(T_{n+1}, T_n, Y_n^k)$ have the most significant computational cost. On the other hand, these fine computations are to be performed in parallel over each interval $[T_n, T_{n+1}]$. In this setting, the algorithm should converge in a number of iterations significantly smaller than the number of time intervals to achieve a real speedup.

In practice, we evaluate the computational time T_{par} of the parareal algorithm as follows. Denote by T_{init} the time required to compute $(Y_n^0)_{n=1,\dots,N}$, by T_{fine} the time required to

compute $(\mathcal{F}(T_{n+1},T_n,Y_n^k))_{n=0,\dots,N-1}$, and by T_{coarse} the time required to compute $\big(\mathcal{G}(T_{n+1},T_n,Y_n^{k+1});\ Y_{n+1}^{k+1}\big)_{n=0,\dots,N-1}$. Then, with an ideal speedup, corresponding to perform the computation of $(\mathcal{F}(T_{n+1},T_n,Y_n^k))_{n=0,\dots,N-1}$ in parallel over N processors, the total time of the parareal run is

$$T_{\text{par}} = T_{\text{init}} + K\Big(\frac{T_{\text{fine}}}{N} + T_{\text{coarse}}\Big), \tag{9}$$

where K is the number of iterations after which we decide to stop the simulation. The value of K can be obtained for example, by imposing a target accuracy to be achieved.

3 The reduced models and their validity

We present in this section the reduced models for the Vlasov-Poisson systems and we assess their validity by numerical experiments. These models will be used in section 4 for the coarse solving in the parareal framework.

3.1 The two-scale limit model

In section 4 of [10], a limit model of the Vlasov-Poisson equation is derived for $\varepsilon \to 0$. More precisely, it is proved that over the time interval $[0,T]$ the solution $(f_\varepsilon, E_\varepsilon)$ to (1)-(2)-(3) two-scale converges to some functions $(F,\mathcal{E})$. These limit functions depend additionally on the fast time scale $\tau \in [0,2\pi]$: $F = F(t,\tau,r,v)$ and $\mathcal{E} = \mathcal{E}(t,\tau,r)$. Moreover, by virtue of [10, Theorem 4.5], we have that there exists $G = G(t,q,u)$ such that

$$F(t,\tau,r,v) = G\big(t,\ \cos(\tau)r - \sin(\tau)v,\ \sin(\tau)r + \cos(\tau)v\big), \tag{10}$$

and $(G,\mathcal{E})$ is the solution of the following system

$$\begin{cases} \dfrac{\partial G}{\partial t} + \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} -\sin(\tau)\Big[\mathcal{E}\big(t,\tau,\cos(\tau)q+\sin(\tau)u\big) + H(\tau)\big(\cos(\tau)q+\sin(\tau)u\big)\Big]d\tau\,\dfrac{\partial G}{\partial q} \\ \qquad + \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} \cos(\tau)\Big[\mathcal{E}\big(t,\tau,\cos(\tau)q+\sin(\tau)u\big) + H(\tau)\big(\cos(\tau)q+\sin(\tau)u\big)\Big]d\tau\,\dfrac{\partial G}{\partial u} = 0, \\ G(0,q,u) = f_0(q,u), \\ \dfrac{1}{r}\dfrac{\partial(r\mathcal{E})}{\partial r} = \Upsilon, \quad \Upsilon(t,\tau,r) = \displaystyle\int_{\mathbb{R}} G\big(t,\ \cos(\tau)r - \sin(\tau)v,\ \sin(\tau)r + \cos(\tau)v\big)dv. \end{cases} \tag{11}$$

Remark 3.1.

(i) The two-scale convergence of $(f_\varepsilon)_\varepsilon$ towards F means that f_ε is approximated by $f_\varepsilon(t,r,v) \approx F(t,t/\varepsilon,r,v)$, when ε vanishes. Note that the function F is obtained from G according to (10)*.*

(ii) The transport equation on G is free of high oscillations and its advection field (the integral terms in (11)*) is obtained by averaging with respect to the rapid time variable the limit electric field $\mathcal{E}$.*

(iii) Note that the limit Poisson equation in (11) *needs to be solved for every* $\tau \in [0, 2\pi]$ *in order to provide the terms* $\mathcal{E}(\cdot, \tau, \cdot)$ *required in the first two equations.*

(iv) The integral terms containing $\mathcal{E}$ *in equation* (11) *have to be approximated, whereas the terms with* H *may be easily computed. As an example, when* H *is given by* (4)*, we have*

$$\frac{1}{2\pi}\int_0^{2\pi} \sin(\tau)\cos^2(\tau)\big(\cos(\tau)q + \sin(\tau)u\big)d\tau = \frac{1}{8}u$$
$$\frac{1}{2\pi}\int_0^{2\pi} \cos(\tau)\cos^2(\tau)\big(\cos(\tau)q + \sin(\tau)u\big)d\tau = \frac{3}{8}q$$

and the two terms in the equation (11) *change accordingly.*

(v) As announced before, the two choices for the function H *(Cases I and II) are sufficient for illustrating the typical averaged behaviour of the solutions, since these choices summarize the general forms of the limit model in* (11)*. More precisely, they entail the possibilities of the non-resonant case (corresponding to* $H \equiv 0$*) and the resonant one (corresponding to* $H(\tau) = \cos^2(\tau)$*); see Theorem 4.5 in [10].*

The aim of this work is to use the reduced model (11) as a coarse solver in the parareal algorithm. Therefore, as one can deduce from (8), we have to apply the limit model with an initial condition starting at a time which is not zero, unlike the one in (11). However, this model is not originally proved to approximate the problem (1)-(2)-(3) over an interval $[s, s+\Delta t]$ when $s \neq 0$. Hence, in order to proceed, we derive below a new model to be used when the initial condition is $G(s, q, u) = f_s(q, u)$ with $s \neq 0$. Note that this issue appears only in the case where $H \neq 0$ contains time-dependent terms, *e.g.*, in Case II. In such an instance, the idea is simply to shift in time the term H by s. Specifically, assume that we have to solve the equation

$$\frac{\partial f_\varepsilon}{\partial t} + \frac{1}{\varepsilon}v\frac{\partial f_\varepsilon}{\partial r} + \Big(E_\varepsilon + \Xi_\varepsilon\Big)\frac{\partial f_\varepsilon}{\partial v} = 0, \quad f_\varepsilon(t = s, r, v) = f_s(r, v), \quad s > 0, \tag{12}$$

together with (2) and (3). Then, by changing the time variable in $\hat{t} = t - s$ and the unknowns *via* $\hat{f}_\varepsilon(\hat{t}, r, v) = f_\varepsilon(t, r, v)$ and $\hat{E}_\varepsilon(\hat{t}, r) = E_\varepsilon(t, r)$, we are left with solving equation (1) in $\hat{t}$ instead of t, with the initial condition $\hat{f}_\varepsilon(\hat{t} = 0, r, v) = f_s(r, v)$ and the term $\widehat{H}(\hat{t}/\varepsilon)$ where $\widehat{H}(\tau) = H(\tau + s/\varepsilon)$. Now, on the basis of the result in [10] for these new unknowns, we replace in the limit equation (11) the term $H(\tau)$ by $H(\tau + s/\varepsilon)$.

Thus, we propose the following reduced model, depending on ε through H:

$$\begin{cases} \dfrac{\partial G_\varepsilon}{\partial t} + \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} -\sin(\tau)\Big[\mathcal{E}\big(t, \tau, \cos(\tau)q + \sin(\tau)u\big) + H\Big(\tau + \frac{s}{\varepsilon}\Big)\big(\cos(\tau)q + \sin(\tau)u\big)\Big]d\tau\, \dfrac{\partial G_\varepsilon}{\partial q} \\ \qquad + \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} \cos(\tau)\Big[\mathcal{E}\big(t, \tau, \cos(\tau)q + \sin(\tau)u\big) + H\Big(\tau + \frac{s}{\varepsilon}\Big)\big(\cos(\tau)q + \sin(\tau)u\big)\Big]d\tau\, \dfrac{\partial G_\varepsilon}{\partial u} = 0, \\ G_\varepsilon(s, q, u) = f_s(q, u), \\ \dfrac{1}{r}\dfrac{\partial(r\mathcal{E})}{\partial r} = \Upsilon, \quad \Upsilon(t, \tau, r) = \displaystyle\int_{\mathbb{R}} G_\varepsilon\big(t,\, \cos(\tau)r - \sin(\tau)v,\, \sin(\tau)r + \cos(\tau)v\big)dv. \end{cases} \tag{13}$$

The function G_ε given by (13) will be used in section 4 as an approximation of f_ε given by (12), when $\varepsilon \to 0$. More precisely, on the basis of (10), we formally write that for $t \in [s, s+\Delta t]$,

$$f_\varepsilon(t,r,v) \approx G_\varepsilon\big(t, \cos(t/\varepsilon)r - \sin(t/\varepsilon)v, \sin(t/\varepsilon)r + \cos(t/\varepsilon)v\big),$$

when ε vanishes. If $s = 0$, we recover the result in [10] on the two-scale limit G.
Now, if we consider the function H in (4), we have for any $S \in \mathbb{R}$,

$$\begin{aligned}
&\frac{1}{2\pi}\int_0^{2\pi} \sin(\tau)\cos^2(\tau+S)\,\big(\cos(\tau)q + \sin(\tau)u\big)d\tau = \frac{1}{8}\Big[(1+2\sin^2(S))u - \sin(2S)q\Big],\\
&\frac{1}{2\pi}\int_0^{2\pi} \cos(\tau)\cos^2(\tau+S)\,\big(\cos(\tau)q + \sin(\tau)u\big)d\tau = \frac{1}{8}\Big[(1+2\cos^2(S))q - \sin(2S)u\Big],
\end{aligned} \tag{14}$$

and thus, the equations for the reduced model of the Vlasov-Poisson Case II over the time interval $[s, s+\Delta t]$ become

$$\left\{\begin{array}{l}
\dfrac{\partial G_\varepsilon}{\partial t} + \left\{\dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} -\sin(\tau)\,\mathcal{E}\big(t,\tau,\cos(\tau)q+\sin(\tau)u\big)d\tau \right.\\
\qquad\qquad \left. -\dfrac{1}{8}\Big[(1+2\sin^2(s/\varepsilon))u - \sin(2s/\varepsilon)q\Big]\right\}\dfrac{\partial G_\varepsilon}{\partial q}\\
\quad + \left\{\dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} \cos(\tau)\,\mathcal{E}\big(t,\tau,\cos(\tau)q+\sin(\tau)u\big)d\tau \right.\\
\qquad\qquad \left. +\dfrac{1}{8}\Big[(1+2\cos^2(s/\varepsilon))q - \sin(2s/\varepsilon)u\Big]\right\}\dfrac{\partial G_\varepsilon}{\partial u} = 0,\\
G_\varepsilon(s,q,u) = f_s(q,u),\\
\dfrac{1}{r}\dfrac{\partial(r\mathcal{E})}{\partial r} = \Upsilon, \quad \Upsilon(t,\tau,r) = \displaystyle\int_{\mathbb{R}} G_\varepsilon\big(t,\,\cos(\tau)r - \sin(\tau)v,\,\sin(\tau)r+\cos(\tau)v\big)dv.
\end{array}\right. \tag{15}$$

In section 3.3, we show numerically that the functions G_ε defined by (15) provide a good approximation of the functions f_ε given by (12)-(2)-(3), when $\varepsilon \to 0$. A theoretical analysis of this approximation goes beyond the scope of this work and might be considered as a future research direction.

3.2 The numerical solving of the reduced models

We solve both systems (1)-(2)-(3) and (15) by a particle-in-cell method [3, 17]. The corresponding unknown distribution functions, denoted by f_ε and G respectively, are approximated by Dirac sums, *i.e.*, by a number N_p of numerical particles. Then, for each unknown, we compute the trajectories of the particles by solving the characteristic curves of the corresponding Vlasov equation, while the self-consistent electric fields are computed on a grid in the physical space by solving the corresponding Poisson equation. This is done in a standard way.

Next, we detail the algorithm we implemented for solving (15), following [10]. The solution G is approximated by

$$G_N(t,q,u) = \sum_{j=1}^{N_p} w_j \delta(q - Q_j(t))\,\delta(u - U_j(t)), \tag{16}$$

where $\big(Q_j(t), U_j(t)\big)$ is the position in the phase space of the macro-particle j moving along a characteristic curve of the Vlasov equation in (15). The problem consists therefore in finding the positions and velocities $\big(Q_j^{n+1}, U_j^{n+1}\big)$ at time $t_{n+1} = (n+1)\Delta t$ from their values at time $t_n = n\Delta t$ by solving the ordinary differential equation

$$\begin{cases} \dfrac{\mathrm{d}Q_j}{\mathrm{d}t} = -\dfrac{1}{8}U_j - \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} \sin(\tau)\, \mathcal{E}\big(t, \tau, \cos(\tau)Q_j + \sin(\tau)U_j\big)d\tau, & Q_j(t_n) = Q_j^n, \\ \dfrac{\mathrm{d}U_j}{\mathrm{d}t} = \quad \dfrac{3}{8}Q_j + \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} \cos(\tau)\, \mathcal{E}\big(t, \tau, \cos(\tau)Q_j + \sin(\tau)U_j\big)d\tau, & U_j(t_n) = U_j^n, \end{cases} \tag{17}$$

where, we considered the equations in the case $s = 0$ for the simplicity of the presentation. Generalization to $s \neq 0$ can be easily derived from (15). Then, we approximate the integral in the fast variable τ by the trapezoidal rule which yields accurate results with a few quadrature points over $[0, 2\pi]$. Thus, if we consider a grid with M cells for $[0, 2\pi]$ and nodes $\tau_\ell = 2\pi\ell/M$ for all $\ell \in \{0, 1, \ldots, M\}$, we are left with solving the differential system

$$\begin{cases} \dfrac{\mathrm{d}Q_j}{\mathrm{d}t} = -\dfrac{1}{8}U_j - \dfrac{1}{M}\displaystyle\sum_{\ell=1}^{M-1} \sin(\tau_\ell)\mathcal{E}\big(t, \tau_\ell, \cos(\tau_\ell)Q_j + \sin(\tau_\ell)U_j\big), & Q_j(t_n) = Q_j^n, \\ \dfrac{\mathrm{d}U_j}{\mathrm{d}t} = \quad \dfrac{3}{8}Q_j + \dfrac{1}{M}\displaystyle\sum_{\ell=1}^{M} \cos(\tau_\ell)\mathcal{E}\big(t, \tau_\ell, \cos(\tau_\ell)Q_j + \sin(\tau_\ell)U_j\big), & U_j(t_n) = U_j^n. \end{cases} \tag{18}$$

In practice, we found that $M = 23$ is a good trade-off value between the computational cost and the accuracy of the integral approximation. In (18), the two-scale electric field $\mathcal{E}$ is to be computed at a fixed time t from the particles $\big(Q_j(t), U_j(t)\big)_{j=1,\ldots,N_p}$ by means of the Poisson equation (15). More precisely, we consider the following steps:

Algorithm 3.2.

1. *For each $\ell \in \{0, 1, \ldots, M\}$, rotate all the particles by the angle τ_ℓ, i.e., compute the positions*

$$\forall \ell \in \{0, 1, \ldots, M\},\ P_j(\tau_\ell) = \cos(\tau_\ell)Q_j + \sin(\tau_\ell)U_j \quad \forall j \in \{1, \ldots, N_p\}.$$

2. *Accumulate particles $\big(P_j(\tau_\ell)\big)_j$ on the grid of $[r_{\min}, r_{\max}]$, in order to derive an approximation of the right-hand side $\Upsilon(t, \tau_\ell, \cdot)$ of the Poisson equation, on the physical grid.*

3. *Solve the Poisson equation in* (15) *by a numerical scheme to obtain the grid electric field $\mathcal{E}(t, \tau_\ell, \cdot)$.*

We finally describe the numerical algorithm for solving the reduced model (15) by the particle-in-cell method. We couple the solving of the ODEs in (18) by an explicit high-order scheme with that of the Poisson equation by the trapezoidal formula for the r-integral.

Algorithm 3.3.

1. *Initialization: Generate random particles $(Q_j, U_j)_j$ following a given distribution function and compute the initial two-scale electric field on the grid, by Algorithm 3.2.*
2. *For every $n \in \{0, 1, \dots\}$, consider (Q_j^n, U_j^n) given at time $t_n = n\Delta t$. Compute (Q_j^{n+1}, U_j^{n+1}) as follows.*
   - *compute the grid electric field following Algorithm 3.2 at time t_n: $\mathcal{E}^n(\tau_\ell, \cdot)$.*
   - *by interpolation compute the electric field in the particles $\cos(\tau_\ell)Q_j^n + \sin(\tau_\ell)U_j^n$.*
   - *push particles (Q_j^n, U_j^n) with a time-stepping scheme for solving* (18).

Note that we can recover the particles $(R_j^n, V_j^n)_j$ approximating the solution f_ε to the system (1)-(2)-(3) at any time t_n, by computing

$$\begin{cases} R_j^n = \cos\Big(\dfrac{t_n}{\varepsilon}\Big)Q_j^n + \sin\Big(\dfrac{t_n}{\varepsilon}\Big)U_j^n \\ V_j^n = -\sin\Big(\dfrac{t_n}{\varepsilon}\Big)Q_j^n + \cos\Big(\dfrac{t_n}{\varepsilon}\Big)U_j^n, \end{cases} \tag{19}$$

as a consequence of Remark 3.1(iii).

We conclude this part with the numerical parameters. In the following, we consider as an initial condition the discontinuous distribution

$$f_0(r, v) = \frac{n_0}{\sqrt{2\pi}\, v_{\rm th}} \exp\Big(-\frac{v^2}{2v_{\rm th}^2}\Big)\chi_{[r_{\min}, r_{\max}]}(r), \tag{20}$$

where, as in [10], $v_{\rm th} = 0.0727518214392$, $r_{\max} = 1.83271471003$ and $r_{\min} = -r_{\max}$ and where $\chi_{[r_{\min}, r_{\max}]}(r) = 1$ if $r \in [r_{\min}, r_{\max}]$ and $\chi_{[r_{\min}, r_{\max}]}(r) = 0$ otherwise. We implement this distribution function with $N_p = 10000$ particles and weights $w_j = (r_{\max} - r_{\min})/N_p$. For the accumulation step, we compute the density ρ_ε on the spatial grid by using first-order B-splines as convolution kernel.
The Poisson equation in (3) is solved as follows (we drop the time variable in E_ε): for $r \in [0, r_{\max}]$ we consider

$$E_\varepsilon(r) = \frac{1}{r}\int_0^r s\rho_\varepsilon(s)\, ds$$

and for $r \in [r_{\min}, 0]$, we define $E_\varepsilon(r) = -E_\varepsilon(-r)$. Then we approximate the integral above by the trapezoidal quadrature with 128 cells. Finally, in order to evaluate the self-consistent electric field in a particle, we use the same B-splines as in the step 2 of Algorithm 3.2, which amounts to an interpolation over the grid nodes. As for the time integration, an explicit Runge-Kutta 4 scheme for both the reduced and the original model is implemented. In addition, when solving the characteristics of equation (1), we use a time step $\delta t = 2\pi\varepsilon/100$, which is sufficiently small for ensuring an accurate solving of the fastest time scale $P = 2\pi\varepsilon$ among the three.

3.3 Validity of the reduced models

The results in [10] do not allow to assess the validity of the reduced models over time intervals of length $1/\varepsilon$ or more. Therefore, we check now numerically for how long these models can be used as accurate approximations of the initial stiff models. We thus compute at any time t_n the following mean relative error

$$\text{Error}(t_n) = \frac{1}{N_p} \sum_{j=1}^{N_p} \frac{\|(R_j^n, V_j^n) - (\widetilde{R}_j^n, \widetilde{V}_j^n)\|_2}{\|(\widetilde{R}_j^n, \widetilde{V}_j^n)\|_2}, \tag{21}$$

where $\|\cdot\|_2$ stands for the Euclidean norm in $\mathbb{R}^2$ and $(R_j^n, V_j^n)_j$ and $(\widetilde{R}_j^n, \widetilde{V}_j^n)_j$ are obtained following Algorithm 3.3 as the numerical solutions at time t_n of the reduced model and of the original model, respectively. Considering the Vlasov-Poisson Case I and the limit model by equation (11), we plot in Fig. 3, left panel, the L^2 norm in time over $[0, T = 64]$ of the mean error defined in (21), for several macro time steps Δt and several values of ε. In order to give more details about the behaviour of the mean relative error, we plot its evolution as obtained with a fixed time step Δt in Fig. 3, right panel. The same curves are plotted in Fig. 4 for the Vlasov-Poisson Case II. We deduce from the left panels that when varying the time step of the limit model in the set $\{0.4, 0.6, 0.8, 1\}$, there is no significant improvement of the accuracy of approximation, in all the cases but Case II with $\varepsilon = 0.001$. Therefore, in the remainder of the paper, the typical values of the coarse time step are 0.8 and 1.

For both Vlasov-Poisson cases, the common observation is that the limit model turns out to be a very accurate approximation of the initial model for every ε and for some small (of order 1) final time. Then, for larger final times (of order 10) the accuracy of the approximation is satisfactory when ε is not bigger than 0.01, whereas for values of ε in $\{0.05, 0.1\}$ the approximation error becomes much larger. Finally, in long runs as $T = 64$, for every value of ε the error of the reduced model is more than 20% in Case I and respectively 50% in Case II. Therefore, the approximation given by the limit model is not accurate anymore. In the next section, we enhance the accuracy of the numerical approximations by means of the parareal algorithm, without a prohibitive computational cost. We conclude this part by underlining the following numerical rate of convergence, well illustrated by the right panels of Figs. 3&4: the reduced model provides a much better approximation to the initial model for the three smallest values of $\varepsilon \in \{0.001, 0.005, 0.01\}$ than for the values $\varepsilon \in \{0.05, 0.1\}$.

Finally, we show that the model proposed in (15) is an accurate approximation of the solution of the Vlasov-Poisson Case II. We thus plot in Fig. 5 the evolution in time of the mean relative error of the limit model in (11) and that of the model in (15). Whereas the first error is computed in a standard manner, the second one is computed as follows: for every $n = 1, 2, \ldots$, in order to obtain the solution at each time $t_n = n\Delta t$, we solve the model during one time step taking as initial condition the solution of the model computed at the previous step t_{n-1}. This is possible with the model in (15) but not with the two-scale limit model which cannot start with an arbitrary initial time. We observe that the errors are very close, for each value of ε, meaning that the model given in (15) provide an approximation of the stiff Vlasov-Poisson solution which is as accurate as the one obtained from the two-scale limit in (11).

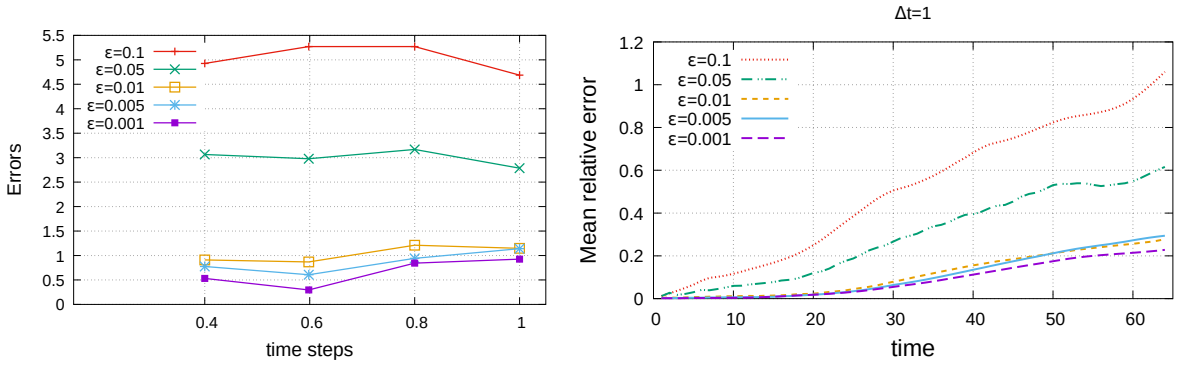

Figure 3: The Vlasov-Poisson Case I for several values of ε. *Left panel:* the discrete L^2 norm on $[0, T = 64]$ of the mean relative error of the two-scale limit model, computed using several time steps Δt. *Right panel:* evolution in time of the mean relative error of the limit model with time step $\Delta t = 1$.

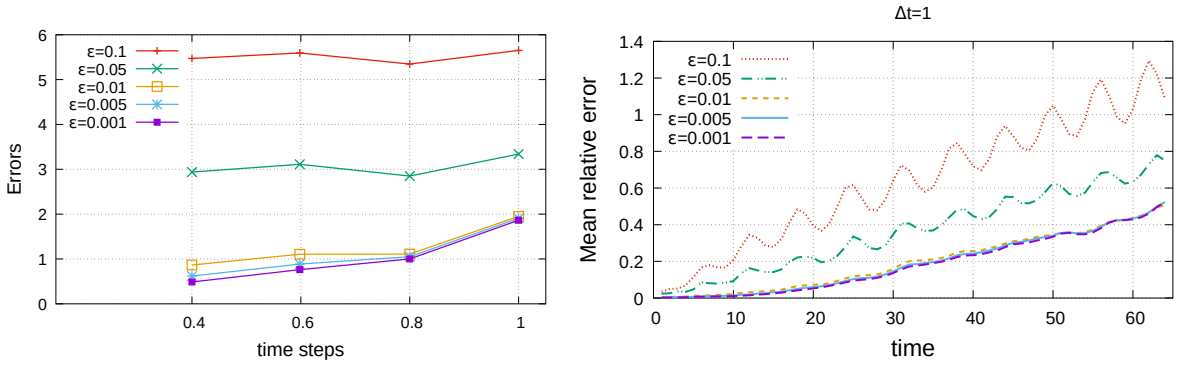

Figure 4: The Vlasov-Poisson Case II for several values of ε. *Left panel:* the discrete L^2 norm on $[0, T = 64]$ of the mean relative error of the two-scale limit model, computed using several time steps Δt. *Right panel:* evolution in time of the mean relative error of the two-scale limit model with time step $\Delta t = 1$.

4 Parareal numerical results

In the previous section, we analyzed the level of approximation of the reduced models and found that in large times of order $1/\varepsilon$, they are not valid anymore. However, we show hereafter that the parareal framework allows to obtain accurate results in such large times. We thus present in this section numerical experiments illustrating the performance of the algorithm. We recall that our strategy is to use the reduced models described in Section 3.1 to define the coarse solvers. More precisely, we start by implementing the parareal algorithm in (7)-(8) by using the initial condition in (20). Then, for the fine solver $\mathcal{F}$, we use the standard numerical approximation of the system (1)-(2)-(3), while for the coarse solver $\mathcal{G}$, we employ the numerical scheme described in Section 3.2, for solving the problem (15).

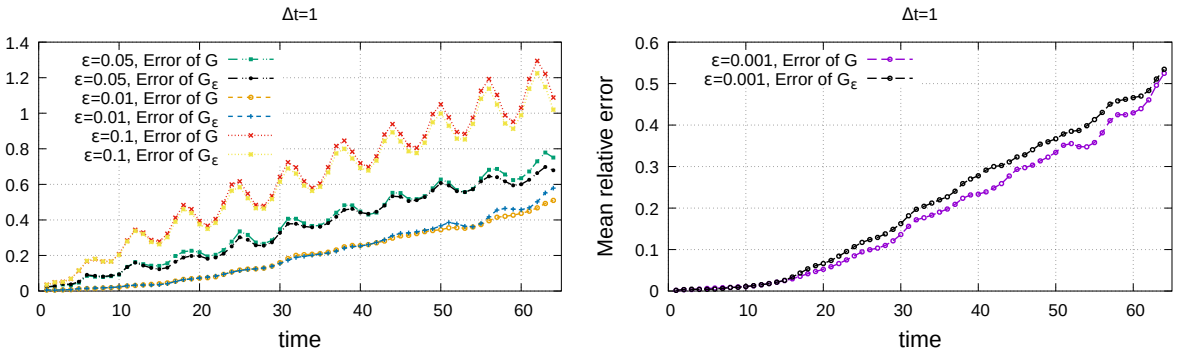

Figure 5: The Vlasov-Poisson Case II for several values of ε: evolution in time of the mean relative error of the two-scale limit model *vs.* the evolution of the error of the model given by G_ε in (15). The time step is $\Delta t = 1$.

The relevant way to evaluate the errors of the parareal algorithm is the following. If we denote by $\mathbb{E}_{\mathcal{G}}$ the error at a fixed final time of the fine solver $\mathcal{F}$ with respect to the exact solution (or in other words the target accuracy, see [20]), then we determine the parareal iteration at which the error $\mathbb{E}_{\rm PA}$ of the algorithm with respect to the exact solution is smaller than $\mathbb{E}_{\mathcal{G}}$.

We start now to discuss the outcome of the numerical simulations. We compute the following mean relative error in $L^\infty([0,T])$ as a function of the number k of parareal iterations

$$\text{Error}(k) = \max_{n\in\{1,\dots,N\}} \left(\frac{1}{N_p} \sum_{j=1}^{N_p} \frac{\|(Y_n^k)_j - \mathcal{X}_j^n\|_2}{\|\mathcal{X}_j^n\|_2} \right), \tag{22}$$

where $\|\cdot\|_2$ stands for the Euclidean norm in $\mathbb{R}^2$, $\left((Y_n^k)_j\right)_j$ are the numerical particles obtained at the k-th iteration of the parareal algorithm and $(\mathcal{X}_j^n)_j$ stand for the numerical particles approximating the initial Vlasov-Poisson model, at time t_n.

For both cases, analytic forms of the solutions are not available and therefore, we need to compute very accurate numerical solutions which play the role of the exact one. In the following, we detail the way we obtain this solution, only for the Vlasov-Poisson Case II. The accuracy is provided by a sufficiently small time step; we keep fixed the numerical parameters for the discretization of the phase space (see the values described in the previous section). We consider as fine solver $\mathcal{F}_1$ the outcome of the Runge-Kutta 4 method applied with a time step $\delta t = 2\pi\varepsilon/100$. Then, we consider the solution obtained with the time step $\widetilde{\delta t} = \delta t/6$ as the reference solution, playing the role of the exact one. Indeed, we plot in Fig. 6 (left panel) the errors of the fine solver with respect to several very fine solutions obtained with time steps $\delta t/2$, $\delta t/4$, $\delta t/6$, $\delta t/8$, $\delta t/10$, and $\delta t/12$. We observe that the error stabilizes when the time step is lower than $\delta t/6$. Therefore, we call in the following the very fine solution the one obtained with the time step $\widetilde{\delta t} = \delta t/6$, as a trade-off between accuracy and very high computational cost. In Fig. 6 we plot only the results at time $T = 20$ but the same behaviour was observed when several final times were used. As described above,

we run the algorithm until its error $\mathbb{E}_{\rm PA}$ is smaller than the target error $\mathbb{E}_{\mathscr{F}_1} = 4.8 \cdot 10^{-4}$ of the fine solver $\mathscr{F}_1$. This is rapidly achieved at iteration 3 (see Fig. 6, right panel). If an improved accuracy is targeted, one can use a finer fine solver $\mathscr{F}$: if one makes use of the solver $\mathscr{F}_2$ with the time step $\delta t = 2\pi\varepsilon/200$, the parareal algorithm achieves an error which is smaller than that of the solver $\mathscr{F}_2$, at iteration 5.

As a first convergence test for the parareal algorithm, we study the property of finding smaller errors with smaller ε, at each parareal iteration, in short-time simulations. We fix the coarse time step to $\Delta t = 0.8$ to keep the same numerical approximation of the reduced model. The final time is $T = 800\varepsilon$ corresponding to approximately 127 rapid rotations P for each ε. We plot the errors of the algorithm in Fig. 7. The curves are relevant, consistent with the property of a better approximation of the reduced model when ε decreases.

Next, we discuss the more interesting long-time behaviour of the parareal algorithm and its convergence properties. The fine solver is fixed from now on to $\mathscr{F}_1$. In Fig. 8, we plot the errors of the algorithm for both Vlasov-Poisson cases, with several ε and final time $T = 64$, while $\Delta t = 0.8$. Firstly, we observe a rapid convergence of the algorithm for the three smallest values of ε. The reason is that the reduced model gives a very good approximation of these solutions, as previously showed in Figs. 3 and 4, right panels. For the values $\varepsilon \in \{0.05, 0.1\}$ the convergence is slower. Similar results were obtained with $\Delta t = 1$. Secondly, in order to get into the details of the efficiency of the parareal algorithm, we display in Table 1 the run times of the (sequential) fine solver $\mathscr{F}_1$ and of the algorithm together with the iterations number K at which $\mathbb{E}_{\rm PA} < \mathbb{E}_{\mathscr{F}}$. We observe that parareal is quite efficient in terms of computational time when ε is small, unlike the cases of $\varepsilon \in \{0.05, 0.1\}$ where the inaccurate approximations of the reduced model require a large number of parareal iterations to achieve the targeted accuracy. We note that the numerical results in Table 1 show that, in some cases, a rather large number of iterations is required to obtain convergence. However, a closer look to Fig. 8 shows that less iterations are needed if our aim is to achieve only the order of the error $\mathbb{E}_{\mathscr{F}}$, *i.e.*, $\mathbb{E}_{\rm PA} \gtrsim \mathbb{E}_{\mathscr{F}}$. For example, in Case II with $\varepsilon = 0.001$, the parareal algorithm converges in 6 iterations, but the order of that error ($\mathbb{E}_{\mathscr{F}} \sim 1.7 \cdot 10^{-2}$) is achieved already at iteration 3, as one can see in the right panel of Fig. 8: the relative difference between these two errors is 0.3%. Thus, if we stop the algorithm at iteration 3, the parareal run time would be smaller. The same comment can be done in Case I with $\varepsilon = 0.01$ and $\varepsilon = 0.1$. This behaviour have also been observed in some cases with the coarse time step $\Delta t = 1$.

Despite the previous remark, the parareal algorithm is not efficient for the two largest values of ε, as expected. Indeed, the coarse solving (*i.e.* the reduced model) is not accurate for those values and therefore, many parareal iterations are required. In order to improve these results, we performed simulations with large values for the coarse time step. This alternative seems promising since the convergence of the parareal algorithm accelerates when Δt increases, or equivalently, when N decreases. We thus plot in Fig. 9 the speedup $S = T_{\rm fine}/T_{\rm par}$ of the algorithm, where $T_{\rm par}$ is given in (9), for $\Delta t \in \{0.8, 1, 2, 4, 8, 16, 32\}$, at the same final time $T = 64$. We note that, unlike in Case I, the largest value for the coarse time step we could consider in Case II is $\Delta t = 8$, since the values 16 and 32 led to very large errors of the reduced model. A first result that we deduce is that a speedup close to 1 is achieved in the cases of $\varepsilon \in \{0.05, 0.1\}$, when $\Delta t = 32$. This is surely not an excellent

result, but a better one than when $\Delta t = 0.8$. Furthermore, for the three smallest values of ε, we found that the choices leading to the best speedups are certainly the time steps $\Delta t = 0.8$ or 1. In addition, in these interesting cases of the stiffest problems, we obtained quite good values of the speedup of the parareal algorithm. At best, we achieved a speedup around 11 for both Vlasov-Poisson cases, for the smallest value of ε, as expected. In some cases the speedup can still be improved as discussed above, namely, by taking into account the convergence of the algorithm together with the achievement of the order of the error. Finally, we report in Tables 2 and 3 the best choices of the time step Δt in terms of speedup, for each value of ε (to be compared to the values obtained with $\Delta t = 0.8$ in Table 1).

We finally study the very long time behaviour of our approximation. In this way, we perform simulations, keeping the coarse time step fixed to $\Delta t = 1$ and increasing the final time with N. Ideally, it would be interesting to compute the error of the parareal algorithm as before, namely, by comparing the approximation with the very fine solution, obtained with the time step $\widetilde{\delta t}$. Since the latter has a computational cost overly expensive, we simply estimate the error by taking as a reference the numerical solution obtained with the time step $\delta t = 2\pi\varepsilon/100$. We consider the Vlasov-Poisson cases with $\varepsilon = 0.01$ that we solve until final times running from 254 to 2037 rapid rotations. We observe from Fig. 10 that the convergence of the parareal algorithm is obtained at $K = 9$ when $N = 16$ or $K = 14$ when $N = 128$ iterations, in Case I. This is an interesting and positive result and we expect to obtain good speedup when the error of the parareal algorithm we aim at is for example of order 10^{-3}.

5 Conclusion

We designed and implemented a specific version of the parareal algorithm to solve some highly oscillating Vlasov-Poisson system. Specifically, we considered for the coarse solving reduced models that we deduced from the two-scale limit of the original Vlasov-Poisson system. In a first step, we estimated numerically this convergence by identifying the smallest values of the parameter ε for which the reduced models provide accurate approximations and we found that in long term runs these approximations become inaccurate, whatever the value of ε is. In order to solve this problem, we investigated the parareal framework. Thus, in a second step, we analyzed the numerical convergence of the parareal algorithm together with its efficiency by determining the coarse time steps leading to the best speedups. We proved that the parareal strategy, unlike the reduced model, has the capability of providing an accurate solution at any final time and for any ε, with a low computational cost. Indeed, we obtained a good speedup when ε is small and the same computational time as that of the fine solution when ε is large.

Acknowledgement: The authors thank Yvon Maday for interesting and helpful discussions about the parareal algorithm.

	$\varepsilon = 0.001$	$\varepsilon = 0.005$	$\varepsilon = 0.01$	$\varepsilon = 0.05$	$\varepsilon = 0.1$
T_{fine} (sec.)	470	94	47	9.4	4.7
T_{par} (sec.)	44.3	13.3	20.5	17.5	62
Parareal iteration K	5	3	6	6	24

	$\varepsilon = 0.001$	$\varepsilon = 0.005$	$\varepsilon = 0.01$	$\varepsilon = 0.05$	$\varepsilon = 0.1$
T_{fine} (sec.)	1043	209	104	20.8	10.4
T_{par} (sec.)	95	22.5	21	24	33.2
Parareal iteration K	6	4	5	8	12

Table 1: The Vlasov-Poisson cases (I at the *top* and II at the *bottom*) for several ε: the computing times in seconds of the fine solver $\mathcal{F}_1$ and of the parareal algorithm until its error is smaller than the error of the fine solver $\mathcal{F}_1$ with respect to the very fine solution. The iterations number at which this error is attained is displayed. The final time is $T = 64$ and the coarse time step fixed to $\Delta t = 0.8$.

	$\varepsilon = 0.001$	$\varepsilon = 0.005$	$\varepsilon = 0.01$	$\varepsilon = 0.05$	$\varepsilon = 0.1$
T_{fine} (sec.)	470	94	47	9.8	4.9
T_{par} (sec.)	44.3	13.3	18	10	5.1
Parareal iteration K/N	5/80	3/80	6/64	2/2	2/2
error $\mathbb{E}_{\mathcal{F}}$	$1.282 \cdot 10^{-2}$	$2.60 \cdot 10^{-3}$	$1.49 \cdot 10^{-3}$	$2.18 \cdot 10^{-3}$	$4.99 \cdot 10^{-3}$

Table 2: The Vlasov-Poisson Case I for several ε: the computing times in seconds of the fine solver $\mathcal{F}_1$ and of the parareal algorithm until its error is smaller than the error $\mathbb{E}_{\mathcal{F}}$ of the fine solver $\mathcal{F}_1$ with respect to the very fine solution. The iterations number at which this error is attained is displayed. The final time is fixed $T = 64$.

	$\varepsilon = 0.001$	$\varepsilon = 0.005$	$\varepsilon = 0.01$	$\varepsilon = 0.05$	$\varepsilon = 0.1$
T_{fine} (sec.)	1041	209	104	21	10.5
T_{par} (sec.)	93	22.5	21	15.5	13
Parareal iteration K/N	5/64	4/80	5/80	6/64	8/8
error $\mathbb{E}_{\mathcal{F}}$	$1.767 \cdot 10^{-2}$	$4.40 \cdot 10^{-3}$	$2.27 \cdot 10^{-3}$	$6.96 \cdot 10^{-3}$	$9.44 \cdot 10^{-3}$

Table 3: The Vlasov-Poisson Case II for several ε: the computing times in seconds of the fine solver $\mathcal{F}_1$ and of the parareal algorithm until its error is smaller than the error $\mathbb{E}$ of the fine solver $\mathcal{F}_1$ with respect to the very fine solution. The iterations number at which this error is attained is displayed. The final time is fixed $T = 64$.

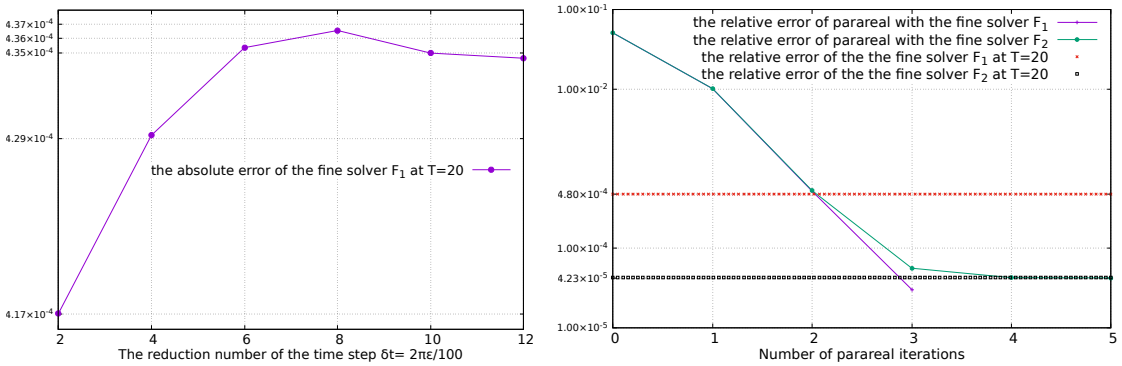

Figure 6: *Left:* The absolute error of the fine solver with respect to several very fine solutions, *i.e.* with several time steps: $\delta t/2, \delta t/4, \ldots, \delta t/12$. *Right:* The relative error with respect to the very fine solution of the parareal algorithm when using two fine solvers: $\mathcal{F}_1$ with $\delta t = 2\pi\varepsilon/100$ and $\mathcal{F}_2$ with $\delta t = 2\pi\varepsilon/200$. The final time is $T = 20$, the coarse time step is $\Delta t = 0.8$ and $\varepsilon = 0.01$.

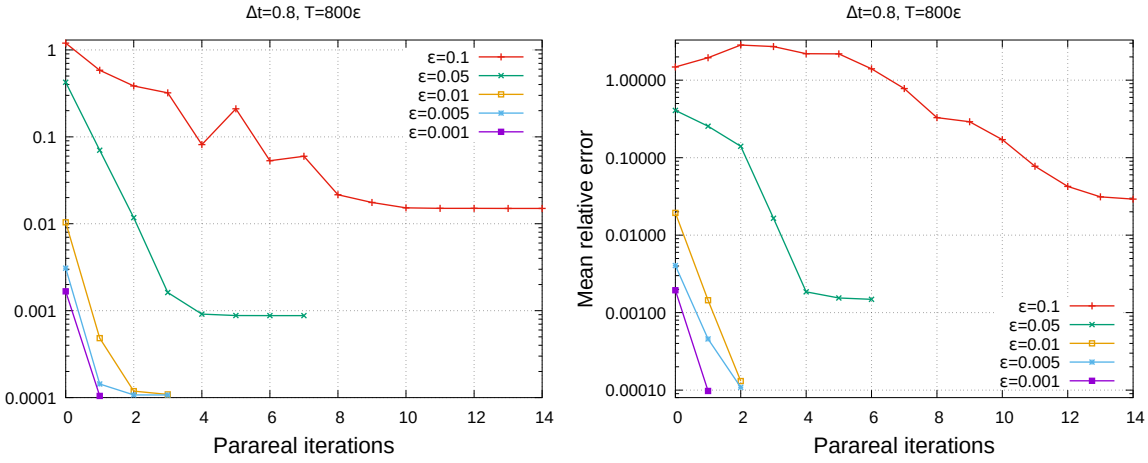

Figure 7: Convergence of the parareal algorithm for the Vlasov-Poisson cases (I at the *left* and II at the *right*) for several ε at final time $T = 800\varepsilon$, with a fixed coarse time step $\Delta t = 0.8$.

References

[1] G. Ariel, S. J. Kim, and R. Tsai. Parareal multiscale methods for highly oscillatory dynamical systems. *SIAM Journal on Scientific Computing*, 38(6):A3540–A3564, 2016.

[2] G. Bal. On the convergence and the stability of the parareal algorithm to solve partial differential equations. In T. J. Barth, M. Griebel, D. E. Keyes, R. M. Nieminen, D. Roose, T. Schlick, R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering*, pages 425–432, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

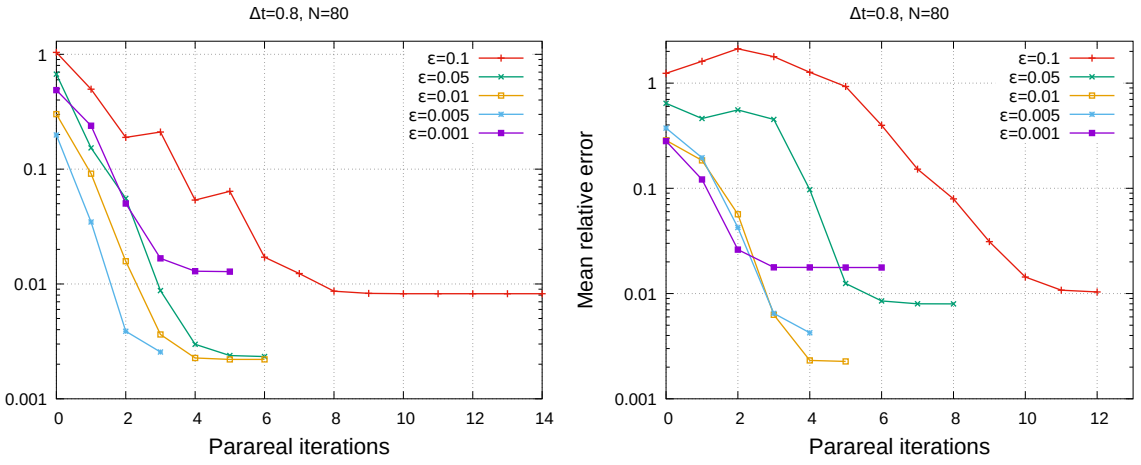

Figure 8: Convergence of the parareal algorithm for the Vlasov-Poisson cases (I at the *left* and II at the *right*) for several ε at final time $T = 64$, with a fixed coarse time step $\Delta t = 0.8$.

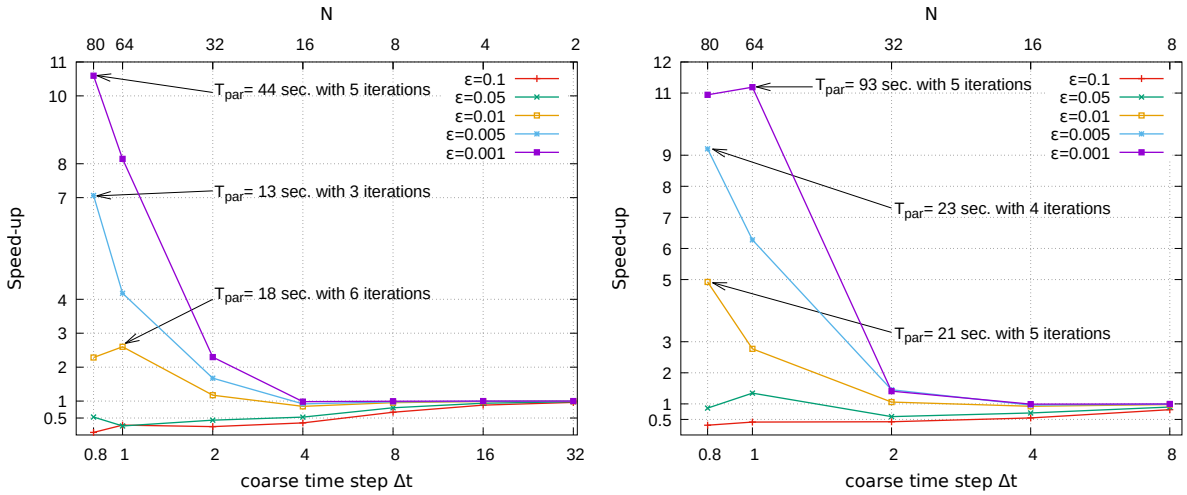

Figure 9: Speedup of the parareal algorithm for the Vlasov-Poisson cases (I at the *left* and II at the *right*), with several coarse time steps at the fixed final time $T = 64$.

[3] C. K. Birdsall and A. B. Langdon. *Plasma Physics via Computer Simulation*. McGraw-Hill, New York, 1985.

[4] J.-P. Chehab and M. Petcu. Parallel matrix function evaluation via initial value ODE modeling. *Computers & Mathematics with Applications*, 72(1):76–91, 2016.

[5] N. Crouseilles, M. Lemou, and F. Méhats. Asymptotic preserving schemes for highly oscillatory Vlasov–Poisson equations. *Journal of Computational Physics*, 248:287–308, 2013.

[6] N. Crouseilles, M. Lemou, F. Méhats, and X. Zhao. Uniformly accurate forward semi-

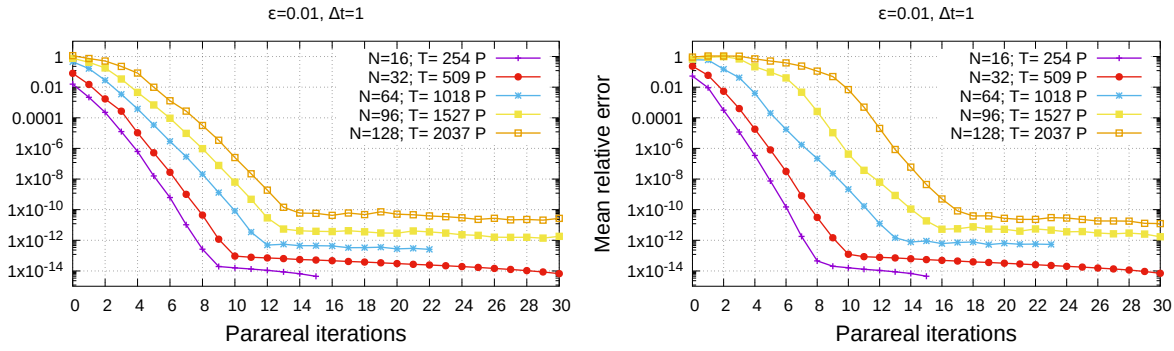

Figure 10: Convergence of the parareal algorithm for the Vlasov-Poisson cases (I at the *left* and II at the *right*), with a fixed coarse time step and an increasing final time T. The fast rotation in time is denoted by $P = 2\pi\varepsilon$.

lagrangian methods for highly oscillatory Vlasov–Poisson equations. *Multiscale Modeling & Simulation*, 15(2):723–744, 2017.

[7] C. Farhat and M. Chandesris. Time-decomposed parallel time-integrators: theory and feasibility studies for fluid, structure, and fluid–structure applications. *International Journal for Numerical Methods in Engineering*, 58(9):1397–1434, 2003.

[8] F. Filbet and E. Sonnendrücker. Modeling and numerical simulation of space charge dominated beams in the paraxial approximation. *Mathematical Models and Methods in Applied Sciences*, 16(05):763–791, 2006.

[9] E. Frénod, S. A. Hirstoaga, and M. Lutz. Long-time simulation of a highly oscillatory Vlasov equation with an exponential integrator. *Comptes Rendus Mécanique*, 342(10-11):595–609, 2014.

[10] E. Frénod, F. Salvarani, and E. Sonnendrücker. Long time simulation of a beam in a periodic focusing channel via a two-scale PIC-method. *Mathematical Models and Methods in Applied Sciences*, 19(02):175–197, 2009.

[11] M. J. Gander and E. Hairer. Nonlinear convergence analysis for the parareal algorithm. In U. Langer, M. Discacciati, D. E. Keyes, O. B. Widlund, and W. Zulehner, editors, *Domain Decomposition Methods in Science and Engineering XVII*, pages 45–56, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[12] M. J. Gander and S. Vandewalle. Analysis of the Parareal Time-Parallel Time-Integration Method. *SIAM Journal on Scientific Computing*, 29(2):556–578, 2007.

[13] I. Garrido, M. S. Espedal, and G. E. Fladmark. A convergent algorithm for time parallelization applied to reservoir simulation. In T. J. Barth, M. Griebel, D. E. Keyes, R. M. Nieminen, D. Roose, T. Schlick, R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering*, pages 469–476, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[14] I. Garrido, B. Lee, G. E. Fladmark, and M. S. Espedal. Convergent iterative schemes for time parallelization. *Mathematics of Computation*, 75(255):1403–1428, 2006.

[15] L. Grigori, S. A. Hirstoaga, V.-T. Nguyen, and J. Salomon. Reduced model-based parareal simulations of oscillatory singularly perturbed ordinary differential equations. *Journal of Computational Physics*, 436:110282, 2021.

[16] T. Haut and B. Wingate. An asymptotic parallel-in-time method for highly oscillatory PDEs. *SIAM Journal on Scientific Computing*, 36(2):A693–A713, 2014.

[17] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. Institute of Physics, Philadelphia, 1988.

[18] J.-L. Lions, Y. Maday, and G. Turinici. A "parareal" in time discretization of PDE's. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, 332:661–668, 2001.

[19] Y. Maday. Parareal in time algorithm for kinetic systems based on model reduction. *High-dimensional partial differential equations in science and engineering, CRM Proc. Lecture Notes*, 41:183–194, 2007.

[20] Y. Maday and O. Mula. An adaptive parareal algorithm. *Journal of computational and applied mathematics*, 377:112915, 2020.

[21] A. Mouton. Two-scale semi-lagrangian simulation of a charged particle beam in a periodic focusing channel. *Kinetic & Related Models*, 2(2):251–274, 2009.

[22] J. A. Sanders, F. Verhulst, and J. Murdock. *Averaging methods in nonlinear dynamical systems*, volume 59. Springer, 2007.

[23] G. A. Staff and E. M. Rønquist. Stability of the parareal algorithm. In T. J. Barth, M. Griebel, D. E. Keyes, R. M. Nieminen, D. Roose, T. Schlick, R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Widlund, and J. Xu, editors, *Domain Decomposition Methods in Science and Engineering*, pages 449–456, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.