SMAI-JCM
SMAI Journal of Computational Mathematics

From particle methods to forward-backward Lagrangian schemes

Martin Campos Pinto & Frédérique Charles

Volume 4 (2018), p. 121-150.
<http://smai-jcm.cedram.org/item?id=SMAI-JCM_2018___4___121_0>

SMAI Journal of Computational Mathematics
Vol. 4, 121-150 (2018)

From particle methods to forward-backward Lagrangian schemes

Martin Campos Pinto [1]
Frédérique Charles [2]

[1] Sorbonne Université, Université Paris-Diderot SPC, CNRS, Laboratoire Jacques-Louis Lions, LJLL, F-75005 Paris
E-mail address: campos@ljll.math.upmc.fr
[2] Sorbonne Université, Université Paris-Diderot SPC, CNRS, Laboratoire Jacques-Louis Lions, LJLL, F-75005 Paris
E-mail address: charles@ljll.math.upmc.fr.

Abstract. In this article we study a novel method for improving the accuracy of density reconstructions based on markers pushed forward by some available particle code. The method relies on the backward Lagrangian representation of the transported density, and it evaluates the backward flow using the current position of point particles seen as flow markers. Compared to existing smooth particle methods with either fixed or transformed shapes, the proposed reconstruction achieves higher locality and accuracy. This is confirmed by our error analysis which shows a theoretical gain of one convergence order compared to the LTP/QTP methods introduced in [8], and by numerical experiments that demonstrate significant CPU gains and an improved robustness relative to the remapping period.

1. Introduction

Particle methods are a popular and efficient tool for the approximation of transport problems. Unfortunately they suffer from weak convergence properties which often prevent an accurate representation of the transported density.

To formalize the problem we consider an abstract transport equation

$$\partial_t f(t,x) + u(t,x)\cdot\nabla f(t,x) = 0, \qquad t\in[0,T], \quad x\in\mathbb{R}^d \tag{1.1}$$

associated with an initial data $f^0:\mathbb{R}^d\to\mathbb{R}$, a final time T and a velocity field $u:[0,T]\times\mathbb{R}^d\to\mathbb{R}^d$. In most cases of interest, the velocity u depends on f through some self-consistent coupling and the problem is non-linear. Here we shall leave this issue aside and assume that u is given and smooth, e.g. $L^\infty(0,T;W^{1;\infty}(\mathbb{R}^d))$, [32], so that there exist characteristic trajectories $X(t)=X(t;s,x)$ solutions to

$$X'(t) = u(t,X(t)), \qquad X(s)=x \tag{1.2}$$

on $[0,T]$, for all $x\in\mathbb{R}^d$ and $s\in[0,T]$. The corresponding flow $F_{s,t}:x\mapsto X(t)$ is then invertible and satisfies $(F_{s,t})^{-1}=F_{t,s}$. In particular, the solution to (1.1) reads

$$f(t,x) = f^0((F_{0,t})^{-1}(x)) \qquad \text{for } \ t\in[0,T],\ x\in\mathbb{R}^d. \tag{1.3}$$

Specifically, as reliable particle solvers do exist for many specific problems, see e.g. [16, 6], we may place ourself in the situation where we are given an accurate solver to push forward arbitrary sets of markers along the forward flow. For simplicity we will assume that the computed flow is exact, and we consider particles centers initially arranged on a Cartesian grid of step size h,

$$x_k^0 = hk, \qquad k\in\mathbb{Z}^d. \tag{1.4}$$

Thus, at time $t^n=n\Delta t$, $n\in\mathbb{N}$, we have at our disposal an unstructured set of particles of the form

$$x_k^n = F_{\rm ex}^{0,n}(x_k^0), \qquad k\in\mathbb{Z}^d, \quad \text{with } \ F_{\rm ex}^{0,n}=F_{0,t^n}$$

and we consider the problem of designing an accurate representation of the transported density $f(t^n)$.

In the standard approach [21, 3, 32], the initial density is first approximated by a weighted collection of smooth "shape functions" centered on the particle positions, of the form

$$f^0_{h,\varepsilon}(x) = \sum_{k\in\mathbb{Z}^d} w^0_k \varphi_\varepsilon(x - x^0_k). \tag{1.5}$$

Here $\varphi_\varepsilon(x) := \varepsilon^{-d}\varphi(\varepsilon^{-1}x)$ is a smooth function with compact support (such as a B-spline or some convolution kernel with vanishing moments [28]) and ε is a smoothing scale which may or may not coincide with h. The weights are defined so that $f^0_{h,\varepsilon}$ approximates f^0 in a measure sense, for instance as $w^0_k = h^d f^0(x^0_k)$, see [32, Sec. I.4], and they evolve according to the differential equation

$$w_k'(t) - (\nabla\cdot u)\left(t, F_{0,t}(x^0_k)\right) w_k(t) = 0, \qquad w_k(0) = w^0_k.$$

The *smooth particle* approximation to $f(t^n)$ is then given by

$$f^{n,\mathrm{sp}}_{h,\varepsilon}(x) = \sum_{k\in\mathbb{Z}^d} w^n_k \varphi_\varepsilon(x - x^n_k) \tag{1.6}$$

with $w^n_k = w_k(t^n)$. Provided some r-th order moment condition on the smoothing kernel φ and assuming $h \lesssim \varepsilon$ (meaning that $h \le C\varepsilon$ for some constant independent of the expressed quantities), the classical error estimate [32, Th. I.5.1] reads

$$\|f(t^n) - f^{n,\mathrm{sp}}_{h,\varepsilon}\|_{L^q} \lesssim \varepsilon^r \|f^0\|_{W^{r,q}} + (h/\varepsilon)^m \|f^0\|_{W^{m,q}}, \quad 1 \le q \le \infty. \tag{1.7}$$

Here we have denoted

$$\|v\|_{W^{r,q}(\omega)} := \|v\|_{L^q(\omega)} + \sum_{s=1}^{r} |v|_{W^{s,q}(\omega)} \qquad |v|_{W^{r,q}(\omega)} := \max_i \Big\{ \sum_{l_1=1}^{d} \cdots \sum_{l_r=1}^{d} \|\partial_{l_1}\cdots\partial_{l_r} v_i\|_{L^q(\omega)} \Big\} \tag{1.8}$$

for functions in Sobolev spaces $W^{r,q}(\omega)$ with $\omega \subset \mathbb{R}^d$, and for conciseness we drop the domain when $\omega = \mathbb{R}^d$. For vectors it will be convenient to use the maximum norm $\|x\|_\infty := \max_i |x_i|$ and the associated $\|A\|_\infty := \max_i \sum_j |A_{i,j}|$ for matrices.

One can improve (1.7) by changing the initial weights w^0_k using better quadrature formulas [12], but in any case such kind of estimates show a weakness of the reconstruction (1.6), namely the need to set $\varepsilon \gg h$ as $\varepsilon, h \to 0$, to guarantee the strong convergence of the approximated densities. As this would lead to a computationally expensive overlapping of particles, in practice many particle codes implement limited values of ε that appear to suffice for the accuracy of the trajectories. In the case where the particles trajectories are exact, the theory indeed guarantees the weak convergence of the approximated densities, independent of ε. Thus in the codes the lack of a sufficient particle overlapping typically translates into strong oscillations in the numerical approximations, the so-called particle noise, see e.g. [30, 11].

To mitigate these oscillations many authors have proposed to use remapping techniques where new weighted particles are periodically computed to approximate the transported density (1.6). The resulting schemes are often referred to as forward semi-Lagrangian (FSL) [18, 24, 29, 14, 17, 27] and they have shown improved convergence rates based essentially on the fact that the frequent reinitializations prevent the particles to become too irregularly distributed. However this has a cost. On the computational level, reinitializing the particles can be expensive and it may introduce numerical diffusion, which conflicts with the conservative essence of the particle method. Advanced techniques have been used to reduce this diffusion, such as high-order non-oscillatory remeshing schemes [27] or multiscale methods, see e.g. [4, 5, 37].

In this article we take a different route to compute non-oscillatory density reconstructions. Following a series of previous works [1, 8, 10, 9], we study a new Lagrangian method that implements in the framework of forward particle methods an improved locality principle proposed by Colombi and Alard in [13] to design highly accurate semi-Lagrangian schemes.

The outline is as follows. In Section 2 we remind how transforming the smooth particle shapes to better follow the characteristic flow allows to reconstruct accurate approximations to the density, at the price of extended particle supports. In Section 3 we then present our new method, for which we provide a priori error estimates in Section 4, and numerical results in Section 5.

2. Accurate particle transport with LTP and QTP approximations

In this section we present the linearly-transformed particle method (LTP) and its second-order extension the quadratically-transformed particle method (QTP) introduced in [8, 10], in order to highlight some of their strengths and drawbacks.

2.1. Using particle shapes with polynomial transformations

Following a natural idea investigated by several authors [23, 2, 20, 12, 15, 1] who considered transforming the smooth particle shapes according to the local variations of the flow, a numerical method to improve the accuracy of the density reconstruction has been proposed and analyzed in [8]. Again the approximated density is obtained as a superposition of weighted smooth particles and (1.5) is used for the initial density, with $\varepsilon = h$. Then, each particle has its own shape $\varphi_{h,k}^n$ that is transported from the initial one $\varphi_{h,k}^0 = \varphi_h$ using a polynomial flow that is defined as a local expansion of the exact backward flow

$$B_{\mathrm{ex}}^{0,n} = (F_{0,t^n})^{-1}.$$

Thus, at the first order the method uses a linearization of $B_{\mathrm{ex}}^{0,n}$ around the k-th particle,

$$B_{h,k,(1)}^{0,n} : x \mapsto x_k^0 + D_k^n(x - x_k^n)$$

with D_k^n an approximation to the $(d \times d)$ backward Jacobian matrix

$$J_{B_{\mathrm{ex}}^{0,n}}(x_k^n) = \left(\partial_j (B_{\mathrm{ex}}^{0,n})_i (x_k^n)\right)_{1\le i,j\le d}. \tag{2.1}$$

This matrix can be computed from the current position of the neighboring particles $x_{k'}^n$, $\|k'-k\|_\infty \le 1$, in two steps: using a finite difference formula on the initial Cartesian particle grid one first obtains an approximation of the forward Jacobian matrix,

$$J_k^n := \left(\frac{\left(x_{k+e_j}^n - x_{k-e_j}^n\right)_i}{2h} \right)_{1\le i,j\le d} \approx J_{F_{\mathrm{ex}}^{0,n}}(x_k^0) \tag{2.2}$$

which is then inverted to compute D_k^n. We refer to Appendix A and [8] for details and a priori error estimates for this procedure. The particle shapes are then obtained by applying the Lagrangian formula (1.3) to the initial shape, using the linearized flow. On time step n this gives

$$\varphi_{h,k}^n(x) = \varphi_h(B_{h,k}^{0,n}(x) - x_k^0) = \varphi_h(D_k^n(x - x_k^n)) \tag{2.3}$$

and the density is reconstructed as a superposition of the resulting *linearly transformed particle* shapes,

$$f_h^{n,\mathrm{ltp}}(x) = \sum_{k\in\mathbb{Z}^d} w_k^n \varphi_h(D_k^n(x - x_k^n)). \tag{2.4}$$

A drawback of this representation is that the matrices D_k^n may have small eigenvalues, which leads to large diameters for the associated particle supports. In practice this means that on a given point the density $f_h^{n,\mathrm{ltp}}$ can have contributions from particles with distant centers x_k^n, as evidenced below on the simple example of the free streaming transport, see Figure 2.1. Clearly this feature deteriorates the locality of the density evaluations and increases their computational cost.

At the second order the local expansion of the backward flow $B^{0,n}_{h,k,(2)}$ takes the form

$$\left(B^{0,n}_{h,k,(2)}(x)\right)_i = (x^0_k)_i + (D^n_k(x-x^n_k))_i + \tfrac{1}{2}(x-x^n_k)^t Q^n_{k,i}(x-x^n_k) \tag{2.5}$$

with $Q^n_{k,i}$ an approximation to the $(d\times d)$ Hessian matrix of the i-th component of the backward flow,

$$H_{(B^{0,n}_{\rm ex})_i}(x^n_k) = \left(\partial_{j_1}\partial_{j_2}(B^{0,n}_{\rm ex})_i(x^n_k)\right)_{1\le j_1,j_2\le d}, \qquad 1\le i\le d, \tag{2.6}$$

Again these matrices can be computed using only the current position of the neighboring particles, $x^n_{k'}$, $\|k'-k\|_\infty \le 1$, see Appendix B. The quadratically transported particle shapes are then defined with the same principle.

However it is necessary to define an a priori support for these particles: because the quadratic mapping $x \mapsto B^{0,n}_{h,k,(2)}(x) - x^0_k$ may vanish far away from x^n_k, the simple expression $\varphi_h(B^{0,n}_{h,k,(2)}(x)-x^0_k)$ has a support that may contain some far away parts, which is obviously not desired since the quadratic expansion (2.5) of $B^{0,n}_{\rm ex}$ is only accurate close to x^n_k. For this reason it is necessary to restrict a-priori the support of the quadratically transformed particles. In [8] these a-priori domains are defined by transporting forward a small extension of the initial particle support using the approximate affine flow. This technical choice allows to prove second-order convergence estimates for the resulting density reconstructions, but it has the disadvantage of further extending the particle support radius as time advances, which further deteriorates the method locality.

Overall, the LTP / QTP approach has resulted in a robust numerical method, and several L^∞ convergence estimates have been derived for the transported densities [8, 9]. However, as noted above it also has the downside that transported particles undergo a stretching of their support that may lead to an important loss in the locality of the computations. Specifically, we see that as time increases the diameter of the particle supports grows like

$$\operatorname{diam}\left(\operatorname{supp}(\varphi^n_{h,k})\right) \approx \operatorname{diam}\left(F^{0,n}_{\rm ex}(\operatorname{supp}(\varphi^0_{h,k}))\right) \approx h|F^{0,n}_{\rm ex}|_{W^{1,\infty}}$$

in the LTP case, which may represent an exponential growth in n. In the QTP method the supports grow even faster, to account for the additional deformations caused by the quadratic terms.

This effect has been experienced in the numerical simulations of higher-dimensional problems such as the 2D2V Vlasov-Poisson system actually implemented in the Selalib platform [33]. It is already visible in the 2D simulations using second order methods presented in this article.

2.2. **The example of free streaming transport**

As an illustration we describe how the above LTP method applies in the simple case of a free streaming transport. Here the problem is posed in a 2D phase space with position and velocity coordinates $z = (x,v) \in \mathbb{R}^2$ corresponding to a one-dimensional physical space. The generalized velocity field associated to a zero acceleration is then $u(x,v) = (v,0)$, and the equation reads

$$\partial_t f(t,x,v) + v\partial_x f(t,x,v) = 0, \qquad t\in[0,T], \quad (x,v)\in\mathbb{R}^2, \tag{2.7}$$

with an initial condition $f(0,\cdot,\cdot) = f^0$. The initial particle approximation f^0_h writes

$$f^0_h(z) = \sum_{k\in\mathbb{Z}^2} w_k \varphi_h(z - z^0_k) \tag{2.8}$$

with particles centers initially located on the Cartesian nodes $z^0_k = h(k_1,k_2) =: (x^0_k, v^0_k)$, $k\in\mathbb{Z}^2$. These centers evolve according to the free-streaming ODE

$$\begin{cases} \dfrac{d}{dt}x_k(t) = v_k(t), & x_k(0) = x^0_k, \\ \dfrac{d}{dt}v_k(t) = 0, & v_k(0) = v^0_k, \end{cases}$$

hence $z_k^n = (x_k^n, v_k^n) = (x_k^0 + t^n v_k^0, v_k^0)$. Here the exact flow $F_{\rm ex}^{0,n}(x,v) = (x + t^n v, v)$ is affine so its linearization around any center z_k^0 is exact: the finite difference approximations (2.2) compute the exact Jacobian matrices $J_k^n = J_{F_{\rm ex}^{0,n}}(z_k^0) = \left(\begin{smallmatrix} 1 & t^n \\ 0 & 1 \end{smallmatrix}\right)$ and the LTP approximation of the density $f(t^n)$ reads

$$f_h^{n,\mathrm{ltp}}(z) = \sum_{k\in\mathbb{Z}^2} w_k \varphi_h\left(D_k^n(z - z_k^n)\right), \qquad \text{with} \qquad D_k^n = (J_k^n)^{-1} = \begin{pmatrix} 1 & -t^n \\ 0 & 1 \end{pmatrix}. \tag{2.9}$$

We then verify that, just as the forward flows, the particle-wise backward flows are exact, i.e., $B_{h,k}^{0,n}(z) = B_{\rm ex}^{0,n}(z)$ for all k. In particular, (2.9) corresponds to an exact transport of the initial particle approximation (2.8).

However, (2.9) also involves a loss of locality that is clearly visible on Figure 2.1: as time advances the particle supports stretch in the x dimension so that the value of $f_h^{n,\mathrm{ltp}}$ on any given point involves contributions from particles whose centers are at distance on the order of $t^n h$.

Of course, locality would be restored by taking $\varepsilon = h$ in the standard particle approximation (1.6),

$$f_h^{n,\mathrm{sp}}(x,v) = \sum_{k\in\mathbb{Z}^2} w_k \varphi_h(z - z_k^n),$$

but as discussed before, this would imply an important loss of accuracy. The gist of the Forward-Backward Lagrangian method will be to allow for local reconstructions without giving up accuracy.

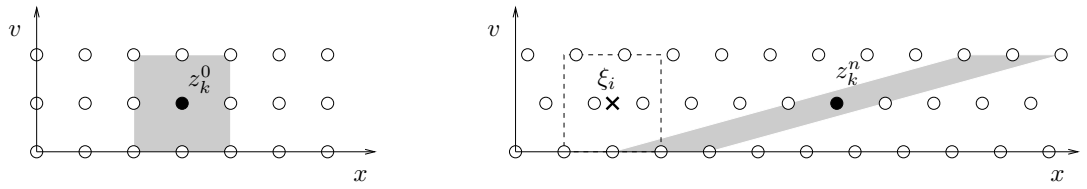


Figure 2.1. Free streaming transport: an initial grid of markers with size h is shown on the left, and on the right at a later time t^n. The shaded area indicates the support of one smooth particle shape function with center represented by a black circle. In the LTP method this shape function is transported along the (exact) linear flow: at $t = t^n$ its support intersects the grid cell represented by a dashed line at a distance $\approx t^n h$ from its center, leading to a loss of locality in the reconstruction of the transported density $f_h^{n,\mathrm{ltp}}$. In the FBL method the reconstruction of the density in the ictured cell involves only nearby particles at distance $\lesssim h$, and locality is preserved.

2.3. **Notations**

We end this section by summarizing the notations used in the article for the different flows and their derivatives:

$F_{\rm ex}^{0,n}$	:	exact forward flow on the time interval $[0, t^n]$
$B_{\rm ex}^{0,n}$	:	exact backward flow on the time interval $[0, t^n]$
$J_{B_{\rm ex}^{0,n}}(x_k^n)$	:	Jacobian matrix of $B_{\rm ex}^{0,n}$ evaluated at x_k^n, the position of the particle k at time t^n
$H_{(B_{\rm ex}^{0,n})_i}(x_k^n)$	:	Hessian matrix of the component i of $B_{\rm ex}^{0,n}$, evaluated at x_k^n
$B_{k,(1)}^{0,n}$	:	linear expansion of $B_{\rm ex}^{0,n}$ around x_k^n (using the exact flow derivatives)
$B_{k,(2)}^{0,n}$	:	quadratic expansion of $B_{\rm ex}^{0,n}$ around x_k^n (using the exact flow derivatives)
D_k^n	:	particle-based approximation of $J_{B_{\rm ex}^{0,n}}(x_k^n)$
$Q_{k,i}^n$	:	particle-based approximation of $H_{(B_{\rm ex}^{0,n})_i}(x_k^n)$
$B_{h,k,(1)}^{0,n}$	:	linear expansion of $B_{\rm ex}^{0,n}$ around x_k^n, using the matrix D_k^n
$B_{h,k,(2)}^{0,n}$	:	quadratic expansion of $B_{\rm ex}^{0,n}$ around x_k^n, using the matrices D_k^n and $Q_{k,i}^n$, $1 \le i \le d$
$B_h^{0,n}$	:	global approximation of $B_{\rm ex}^{0,n}$ used in the FBL reconstruction.

In the remapped version of the methods, the above notations will be extended to time intervals of the form $[t^m, t^n]$ with m the last remapping time step preceding n (see, e.g. Section 3.3).

3. The Forward-Backward Lagrangian (FBL) approximation

Since it is the accurate transport of the smooth particle shapes that causes a loss of locality in the computation of the approximated density, a natural option to restore locality is to follow the elegant approach of [13] and abandon the forward description of f in terms of smooth particle shapes.

Specifically, we opt for a *backward representation of f* based on a *forward description of the flow*. We can indeed recycle one central step of the method above, namely the local approximations to the backward flow using the neighboring particles positions. Instead of using these accurate backward flows to improve the transport of the particles shapes, we will combine them to derive an alternate implementation of the Lagrangian representation of f that is at the basis of standard backward semi-Lagrangian (BSL) methods, see e.g. [34].

The resulting method thus combines a forward part where point particles (markers) are pushed along a standard numerical flow, and a backward part where the transported density is reconstructed with a Lagrangian point of view and an accurate approximation of the backward flow.

3.1. Forward and backward approaches to transport approximation

Before describing in details the FBL method we summarize the main differences between forward and backward approaches for the approximation of transport problems. As written above, *forward* methods approximate the transported density by a superposition of numerical particles (either Dirac masses or smooth shape functions) pushed forward. In the case of smooth particles the simplest option is to translate the particles shapes along with the centers, which leads to a representation of the form (1.6),

$$f_{h,\varepsilon}^{n,\rm fwd}(x) = \sum_{k\in\mathbb{Z}^d} w_k^n \varphi_\varepsilon(x - F_{\rm ex}^{0,n}(x_k^0)). \tag{3.1}$$

The essence of the LTP and QTP methods described in Section 2 was to improve this approximation by transforming each particle shape according to the local variations of the flow.

On the other hand, the *backward* approach exploits the Lagrangian representation formula (1.3) of the transported densities. Starting from an initial approximation f_h^0, this representation reads

$$f_h^{n,\mathrm{bwd}}(x) = f_h^0(B_{\mathrm{ex}}^{0,n}(x)) \tag{3.2}$$

where $B_{\mathrm{ex}}^{0,n} = (F_{\mathrm{ex}}^{0,n})^{-1} : \mathbb{R}^d \to \mathbb{R}^d$ is the backward flow on the time interval $[0, t^n]$. The principle of the subsequent FBL method is then to provide an efficient approximation scheme for this backward flow, given a collection of point markers that have been pushed forward by an accurate particle scheme.

We note in passing that it is also a Lagrangian formula that is at the basis of the improved particle transport in the LTP/QTP method. Indeed if f_h^0 is a collection of smooth particles of scale $\varepsilon = h$, then (3.2) becomes

$$f_h^{n,\mathrm{bwd}}(x) = \sum_{k\in\mathbb{Z}^d} w_k^n \varphi_h(B_{\mathrm{ex}}^{0,n}(x) - x_k^0) \tag{3.3}$$

and the LTP/QTP method essentially consists in implementing this formula with approximate backward flows, one per numerical particle.

Finally we observe that in the case of a constant velocity $u(t,x) = \overline{u}$, the forward flow reads $F_{\mathrm{ex}}^{0,n}(y) = y + \overline{u}t^n$ and its inverse is $B_{\mathrm{ex}}^{0,n}(y) = y - \overline{u}t^n$, so that (3.1) and (3.3) are equivalent.

3.2. **Description of the method**

The FBL approximation to the exact solution $f(t^n, x) = f^0(B_{\mathrm{ex}}^{0,n}(x))$ consists of the following steps.

(i) To every particle x_k^n we associate a polynomial backward flow $B_{h,k}^{0,n}$ which approximates the exact one close to x_k^n, as in the LTP and QTP methods. We remind that at the first order this flow reads

$$B_{h,k}^{0,n} = B_{h,k,(1)}^{0,n} : x \mapsto x_k^0 + D_k^n(x - x_k^n) \qquad \text{with} \quad D_k^n \approx J_{B_{\mathrm{ex}}^{0,n}}(x_k^n) \tag{3.4}$$

see Appendix A, and at the second order it takes the form

$$B_{h,k}^{0,n} = B_{h,k,(2)}^{0,n} : x \mapsto x_k^0 + D_k^n(x - x_k^n) + \tfrac{1}{2}\big((x - x_k^n)^t Q_{k,i}^n (x - x_k^n)\big)_{1\le i\le d} \tag{3.5}$$

with $Q_{k,i}^n \approx H_{(B_{\mathrm{ex}}^{0,n})_i}(x_k^n)$, $1 \le i \le d$, see Appendix B.

(ii) To smoothly patch these local flows together we then consider a partition of unity

$$\sum_{i\in\mathbb{Z}^d} S(x - i) = 1, \qquad x \in \mathbb{R}^d \tag{3.6}$$

involving a compactly supported, non-negative shape function S (for instance a B-spline), and a grid of step size h. Writing the corresponding nodes as $\xi_i = ih$ to avoid a confusion with the particles positions, the scaled formula reads $\sum_{i\in\mathbb{Z}^d} S_{h,i}(x) = 1$ where $S_{h,i}(x) = S((x - \xi_i)/h)$. A global approximation to the backward flow is then defined as

$$B_h^{0,n}(x) := \sum_{i\in\mathbb{Z}^d} B_{h,k^*(n,i)}^{0,n}(x) S_{h,i}(x) \tag{3.7}$$

where $k^*(n,i)$ is the index of the closest marker to the node ξ_i,

$$k^*(n,i) := \mathrm{argmin}_{k\in\mathbb{Z}^d} \|x_k^n - \xi_i\|_\infty.$$

(iii) The approximate solution is finally obtained by a standard Lagrangian formula involving the initial density

$$f_h^{n,\mathrm{fbl}}(x) := f^0\big(B_h^{0,n}(x)\big). \tag{3.8}$$

Remark 3.1 (improved locality)**.** By comparison with the LTP/QTP method, we see that locality is recovered in the above procedure by using only the closest particle to the nodes of the reconstruction grid. In the case of the free streaming transport illustrated on Figure 2.1, we verify that evaluating the approximated FBL density at an arbitrary point only involves nearby markers, unlike what was observed with the LTP method.

Remark 3.2 (second order flow reconstructions)**.** In the second order case we observe that this scheme is also simpler as the QTP method, as it does not require to estimate the a priori supports of transformed particles.

Remark 3.3 (conservative transport)**.** The FBL approximation can be extended with little extra cost to the case of a transport equation in conservation form

$$\partial_t f(t,x) + \nabla \cdot (uf)\,(t,x) = 0. \tag{3.9}$$

In the case of an incompressible flow ($\nabla \cdot u = 0$), this form is equivalent with (1.1). Otherwise the exact solution to (3.9) reads

$$f(t,x) = f^0((B_{\rm ex}^{0,n}(x))\det(J_{B_{\rm ex}^{0,n}}(x)).$$

Consistent with the spirit of the reconstruction (3.8), an approximation $j_h^{0,n}(x)$ to $\det(J_{B_{\rm ex}^{0,n}}(x))$ is defined by

$$j_h^{0,n}(x) = \sum_{i\in\mathbb{Z}^d} j_{h,i}^{0,n} S_{h,i}(x) \qquad \text{with} \quad j_{h,i}^{0,n} = \det\left(D^n_{k^*(n,i)}\right),$$

and we define the corresponding FBL approximation to the transported density by

$$f_h^{n,\rm fbl}(x) := f^0(B_h^{0,n}(x))j_h^{0,n}(x). \tag{3.10}$$

Remark 3.4. Unlike in the LTP case, we can define the FBL approximate density without an initial discretization of f^0, which corresponds to (3.8). It is also possible to use a grid approximation of the initial density, and this case is considered with the remapped version in Section 3.3 below.

3.3. **Remapped FBL method**

Because the regularity of the characteristic flow deteriorates over time, it is important to remap the particles before the approximated flow becomes too inaccurate. Remapping essentially consists of using the density transported up to some time t^m as the initial data of a new transport problem, so that the relevant flow map is reset to the identity. Specifically we replace the approximate density (3.8) given by the FBL method with a new nodal representation on the grid,

$$f_h^m(x) = \sum_k w_k^m \varphi_h(x - x_k^0) \approx f_h^{m,\rm fbl}(x) \tag{3.11}$$

and for the subsequent time steps we follow a new set of particles arranged on the Cartesian grid (1.4).

To project on the grid we may use tensor products of univariate B-splines defined recursively by $\mathcal{B}_p(x) = \int_{x-\frac12}^{x+\frac12} \mathcal{B}_{p-1}$, $\mathcal{B}_0 := \chi_{[-\frac12,\frac12]}$. The approximation reads then

$$A_h g(x) := \sum_{k\in\mathbb{Z}^d} a_{h,k}(g)\varphi_h(x - x_k^0), \qquad \varphi_h(x) = \prod_{1\le i\le d} \frac1h \mathcal{B}_p\left(\frac{x_i}{h}\right)$$

where $a_{h,k}(g)$ is a coefficient that depends on the values of g on a local stencil around x_k^0, such that polynomials of degree $\ge p$ are exactly reproduced by A_h, see [36, 8], but many other approximations are possible. Here we simply assume that A_h satisfies

$$\|A_h g - g\|_{W^{s,\infty}} \le C_A h^{q-s} |g|_{W^{q,\infty}}, \qquad \text{for } 0 \le s \le q \le p+1 \tag{3.12}$$

for some constant c_A.

Thus, if we denote the remapping steps by $m_1, \dots m_R$ and let $m_0 = 0$ be the initial step where the initial data f^0 is also projected, the remapped-FBL approximation reads

$$f_h^{n,\mathrm{fbl}}(x) = f_h^m(B_h^{m,n}(x)), \qquad m = \max\{m_r < n : r \in \{0, \dots, R\}\} \tag{3.13}$$

with

- a numerical flow
$$B_h^{m,n} = \sum_{i \in \mathbb{Z}^d} B_{h,k^*(n,i)}^{m,n} S_{h,i} \tag{3.14}$$
computed with the FBL method (3.7), using the current particles $x_k^n = F_{\mathrm{ex}}^{m,n}(x_k^0)$,
- a numerical density f_h^m which is either the approximation of the initial data, or that of an approximated solution transported with the FBL method if m is a remapping step, that is,
$$f_h^m = \begin{cases} A_h f^0 & \text{if } m = m_0 = 0 \\ A_h f_h^{m,\mathrm{fbl}} & \text{if } m = m_r > 0. \end{cases} \tag{3.15}$$

We note that if we only consider the remapping time steps the proposed method is formally a backward semi-Lagrangian scheme [35, 34]

$$f_h^{m_{r+1}} = A_h\Big(f_h^{m_r} \circ B_h^{m_r, m_{r+1}}\Big) \tag{3.16}$$

where the approximated backward flow is computed using the forward-pushed markers. Here an interesting feature of the remapped FBL scheme is that the approximate backward flow relies on a forward particles pusher, which in many cases is simpler to implement than backward approximations of characteristic trajectories.

4. A priori error analysis

Following the error analysis established in [8] for the LTP and QTP methods, it is possible to derive a priori convergence rates for the proposed method. Here we take into account the error induced by the particle-based approximation of the Jacobian matrices, and estimates are available provided h is small enough to guarantee that these matrices are invertible, see Appendix. We first consider the case where no remappings are performed.

4.1. A priori estimates for the FBL method without remappings

In the case without remappings we establish a priori estimates for the approximation of the transport alone.

Theorem 4.1. *Let $h \le h^*(F_{\mathrm{ex}}^{0,n})$ as in* (A.5) *and assume $B_{\mathrm{ex}}^{0,n}$ and $F_{\mathrm{ex}}^{0,n} \in W^{2,\infty}(\mathbb{R}^d)$. The first order FBL approximation* (3.8)*, i.e. with local flows* (3.4)*, satisfies*

$$\|f_h^{n,\mathrm{fbl}} - f(t^n)\|_{L^\infty} \le Ch^2 \tag{4.1}$$

with a constant independent of h that is specified in the proof.

Remark 4.2 (improved accuracy)**.** Compared to the error estimates established in [8] for the LTP and QTP methods, Theorems 4.1 above and 4.3 below represent a gain of one order in the convergence rates. The main reason for this gain is the fact that in the LTP and QTP methods, the accuracy of the reconstruction is estimated separately on each of the transported particles shapes, whose Lipschitz constants grow like h^{-1} as h goes to 0. In the FBL method the accuracy of the reconstruction relies instead on the global Lipschitz constant of the remapped densities, which is bounded uniformly in h.

Proof. By direct application of a Taylor expansion we see that the (exact) linearization of the backward flow around some particle x_k^n, namely

$$B_{k,(1)}^{0,n} : x \mapsto x_k^0 + (J_{B_{\rm ex}^{0,n}(x_k^n)})(x - x_k^n)$$

satisfies an priori estimate

$$\|B_{k,(1)}^{0,n}(x) - B_{\rm ex}^{0,n}(x)\|_\infty \leq \frac{1}{2}|B_{\rm ex}^{0,n}|_{W^{2,\infty}}\|x - x_k^n\|_\infty^2. \tag{4.2}$$

Thanks to the assumption $h \leq h^*(F_{\rm ex}^{0,n})$ we can next use Lemma A.1 (specifically, estimate (A.6) with $q = 1$) to estimate the finite difference approximation on the backward flow Jacobian matrix. For the local numerical flow (3.4), we thus have

$$\begin{aligned}\|B_{h,k,(1)}^{0,n}(x) - B_{\rm ex}^{0,n}(x)\|_\infty &\leq \|(D_k^n - J_{B_{\rm ex}^{0,n}(x_k^n)})(x - x_k^n)\|_\infty + \|B_{k,(1)}^{0,n}(x) - B_{\rm ex}^{0,n}(x)\|_\infty \\ &\leq C_F\|x - x_k^n\|_\infty\Big(d^2|F_{\rm ex}^{0,n}|_1^{2(d-1)}h + \frac{1}{2}\|x - x_k^n\|_\infty\Big)\end{aligned} \tag{4.3}$$

with $C_F = \max\left(|B_{\rm ex}^{0,n}|_{W^{2,\infty}}, |F_{\rm ex}^{0,n}|_{W^{2,\infty}}\right)$. Our estimate for the global flow $B_h^{0,n}$ will gather bounds of this form for $k = k^*(n,i)$ with $i \in \mathbb{Z}^d$, and $x \in \mathrm{supp}(S_{h,i})$ where $S_{h,i}(x) := S\left(\frac{1}{h}(x - \xi_i)\right)$. In particular, we need to evaluate the distance between a node ξ_i and its associated particle $x_{k^*(n,i)}^n$, and to this end we note that for an arbitrary $k \in \mathbb{Z}^d$ we have

$$\|x_k^n - \xi_i\|_\infty = \|F_{\rm ex}^{0,n}(x_k^0) - F_{\rm ex}^{0,n}(B_{\rm ex}^{0,n}(\xi_i))\|_\infty \leq |F_{\rm ex}^{0,n}|_{W^{1,\infty}}\|x_k^0 - B_{\rm ex}^{0,n}(\xi_i)\|_\infty.$$

By definition of $k^*(n,i)$ and using the fact that the markers x_k^0 are on a Cartesian grid of step h, this yields

$$\|x_{k^*(n,i)}^n - \xi_i\|_\infty \leq |F_{\rm ex}^{0,n}|_{W^{1,\infty}} \min_{k\in\mathbb{Z}^d}\|x_k^0 - B_{\rm ex}^{0,n}(\xi_i)\|_\infty \leq \frac{h}{2}|F_{\rm ex}^{0,n}|_{W^{1,\infty}}.$$

Writing then $\rho_S = \frac{1}{2}\,\mathrm{diam}(\mathrm{supp}(S))$ we see that the support of $S_{h,i}$ is an ℓ^∞ ball of center ξ_i and radius $h\rho_S$. Thus,

$$x \in \mathrm{supp}(S_{h,i}) \quad \Longrightarrow \quad \|x - x_{k^*(n,i)}^n\|_\infty \leq \|x - \xi_i\|_\infty + \|\xi_i - x_{k^*(n,i)}^n\|_\infty \leq h\rho(S,n)$$

with

$$\rho(S,n) := \rho_S + \frac{1}{2}|F_{\rm ex}^{0,n}|_{W^{1,\infty}}. \tag{4.4}$$

In particular, the bound (4.3) evaluated for $k = k^*(n,i)$ gives

$$\|B_{h,k^*(n,i),(1)}^{0,n} - B_{\rm ex}^{0,n}\|_{L^\infty(\mathrm{supp}(S_{h,i}))} \leq C_1 h^2, \qquad i \in \mathbb{Z}^d \tag{4.5}$$

with

$$C_1 = C_F\rho(S,n)(d^2|F_{\rm ex}^{0,n}|_1^{2(d-1)} + \frac{1}{2}\rho(S,n)). \tag{4.6}$$

Using the partition unity properties of the shape function S, we then write for the global flow

$$(B_h^{0,n} - B_{\rm ex}^{0,n})(x) = \sum_{i\in\mathbb{Z}^d}(B_{h,k^*(n,i),(1)}^{0,n} - B_{\rm ex}^{0,n})(x)S_{h,i}(x) \leq C_1 h^2 \sum_{i\in\mathbb{Z}^d} S_{h,i}(x) = C_1 h^2, \qquad x \in \mathbb{R}^d. \tag{4.7}$$

Using the smoothness of f^0 completes the proof, with $C = C_1|f^0|_{W^{1,\infty}}$. ∎

Theorem 4.3. *Let $h \leq h^*(F_{\rm ex}^{0,n})$ as in* (A.5) *and assume $B_{\rm ex}^{0,n}$ and $F_{\rm ex}^{0,n} \in W^{3,\infty}(\mathbb{R}^d)$. The second order FBL approximation* (3.8)*, i.e. with local flows* (3.5)*, satisfies*

$$\|f_h^{n,\rm fbl} - f(t^n)\|_{L^\infty} \leq Ch^3$$

with a constant that depends on the exact flow $F_{\rm ex}^{0,n}$ and its inverse $B_{\rm ex}^{0,n}$, but not on h.

Proof. The proof is similar to that of Theorem 4.1. We first observe that the (exact) quadratic expansion of the backward flow around some particle x_k^n, namely

$$B_{k,(2)}^{0,n} : x \mapsto x_k^0 + \big(J_{B_{\rm ex}^{0,n}}(x_k^n)\big)(x - x_k^n) + \frac{1}{2}\big((x - x_k^n)^t \big(H_{(B_{\rm ex}^{0,n})_i}(x_k^n)\big)(x - x_k^n)\big)_{1\le i\le d}$$

satisfies an priori estimate

$$\|B_{k,(2)}^{0,n}(x) - B_{\rm ex}^{0,n}(x)\|_\infty \le \frac{1}{6}|B_{\rm ex}^{0,n}|_{W^{3,\infty}}\|x - x_k^n\|_\infty^3. \tag{4.8}$$

Thanks to the assumption $h \le h^*(F_{\rm ex}^{0,n})$ we can next use Lemma A.1 (specifically, estimate (A.6) with $q = 2$) and B.1 to estimate the finite difference approximations for the backward flow Jacobian and Hessian matrices. For the local numerical flow (3.5), we thus have

$$\begin{aligned}\|B_{h,k,(2)}^{0,n}(x) - B_{\rm ex}^{0,n}(x)\|_\infty &\le \|B_{k,(2)}^{0,n}(x) - B_{\rm ex}^{0,n}(x)\|_\infty + \|D_k^n - J_{B_{\rm ex}^{0,n}}(x_k^n)\|_\infty\|(x - x_k^n)\|_\infty \\ &\quad + C\max_i \|Q_{k,i}^n - H_{(B_{\rm ex}^{0,n})_i}(x_k^n)\|_\infty\|(x - x_k^n)\|_\infty^2 \qquad (4.9)\\ &\le C(F_{\rm ex}^{0,n})(\|x - x_k^n\|_\infty^3 + h\|x - x_k^n\|_\infty^2 + \|x - x_k^n\|_\infty h^2)\end{aligned}$$

with a constant depending on the exact flow. The rest of the proof is the same. ∎

Remark 4.4. In the case of the FBL approximation (3.10) of the conservative transport equation, it is possible to prove using the same arguments that if $B_{\rm ex}^{0,n}$ and $F_{\rm ex}^{0,n} \in W^{3,\infty}(\mathbb{R}^d)$ then the first order FBL approximation (3.8) satisfies, under a modified condition on h of the form $h \le h^{**}(F_{\rm ex}^{0,n})$,

$$\|f_h^{n,\rm fbl} - f(t^n)\|_{L^\infty} \le Ch^2$$

with a constant independent of h.

4.2. **A priori estimates for the FBL method with remappings**

Let us denote by $\|A\|_{L^\infty} = \inf_{g\in\mathcal{C}} \frac{\|A_h g\|_{L^\infty}}{\|g\|_{L^\infty}}$ the L^∞ norm of the approximation operator A_h. We have the following estimate for the remapped method described in Section 3.3.

Theorem 4.5. *Assume that f^0, $F_{\rm ex}^{0,n}$ and $B_{\rm ex}^{0,n} \in W^{2,\infty}(\mathbb{R}^d)$. If the remapping time steps are such that $h \le h^*(F^{m_{r-1},m_r})$ for $r = 1,\dots,R$, as in* (A.5)*, then the (first order) remapped FBL scheme* (3.13)-(3.15) *satisfies*

$$\|f_h^{n,\rm fbl} - f(t^n)\|_{L^\infty} \le Ch^2 \tag{4.10}$$

with a constant specified in the proof, that may depend on the number of remappings R but not on h.

Proof. Writing m the last remapping step before n, the remapped FBL approximation reads $f_h^{n,\rm fbl}(x) = f_h^m(B_h^{m,n}(x))$, whereas the exact solution $f^n = f(t^n)$ satisfies $f^n(x) = f^m(B_{\rm ex}^{m,n}(x))$. Thus, we have

$$\begin{aligned}\|f_h^{n,\rm fbl} - f^n\|_{L^\infty} &\le \|f_h^m(B_h^{m,n}) - f^m(B_h^{m,n})\|_{L^\infty} + \|f^m(B_h^{m,n}) - f^m(B_{\rm ex}^{m,n})\|_{L^\infty} \\ &\le \|f_h^m - f^m\|_{L^\infty} + |f^m|_{W^{1,\infty}}\|B_h^{m,n} - B_{\rm ex}^{m,n}\|_{L^\infty}.\end{aligned} \tag{4.11}$$

Since the flow $B_h^{m,n}$ is obtained by applying the FBL method on the particles transported from t^m to t^n, the arguments used above to establish the bound (4.7) give here

$$\|B_h^{m,n} - B_{\rm ex}^{m,n}\|_{L^\infty} \le C_{m,n}h^2$$

with $C_{m,n} = |F^{m,n}|_{W^{2,\infty}}\rho(S, n-m)\big(d^2|F_{\rm ex}^{m,n}|_1^{2(d-1)} + \frac{1}{2}\rho(S, n-m)\big)$ and $\rho(S,\cdot)$ is defined in (4.4). As for the remapped approximation error at time t^m, it satisfies

$$\begin{aligned}\|f_h^m - f^m\|_{L^\infty} &\le \|A_h(f_h^{m,\rm fbl} - f^m)\|_{L^\infty} + \|(A_h - I)f^m\|_{L^\infty} \\ &\le \|A_h\|_{L^\infty}\|f_h^{m,\rm fbl} - f^m\|_{L^\infty} + C_A|f^m|_{W^{2,\infty}}h^2.\end{aligned}$$

Gathering the above estimates thus yields

$$\|f_h^{n,\mathrm{fbl}} - f(t^n)\|_{L^\infty} \leq \|A_h\|_{L^\infty}\|f_h^{m,\mathrm{fbl}} - f^m\|_{L^\infty} + h^2\Big(C_A|f^m|_{W^{2,\infty}} + C_{m,n}|f^m|_{W^{1,\infty}}\Big).$$

Denoting for convenience by $m_{R+1} = N$ the last time step (where no remapping is actually performed) we then observe that the error term defined by $e_r := \max_{m_{r-1}<n\leq m_r}\|f_h^{n,\mathrm{fbl}} - f(t^n)\|_{L^\infty}$ for $r \geq 1$ and $e_0 := 0$ satisfies a recursive bound

$$e_r \leq \|A_h\|_{L^\infty} e_{r-1} + \beta h^2, \qquad r = 1, \dots R+1,$$

where we have denoted

$$\beta = C_A\|f\|_{L^\infty([0,T];W^{2,\infty})} + \max_{1\leq r\leq R+1}\max_{m_{r-1}<n\leq m_r}\big(|f^{m_{r-1}}|_{W^{1,\infty}}C_{m_{r-1},n}\big).$$

This gives $e_r \leq \alpha_r\beta h^2$ with $\alpha_r = (\|A_h\|_{L^\infty}^r - 1)/(\|A_h\|_{L^\infty} - 1)$ if $\|A_h\|_{L^\infty} > 1$ and $\alpha_r = r$ if $\|A_h\|_{L^\infty} = 1$. In particular this implies

$$\|f_h^{n,\mathrm{fbl}} - f^n\|_{L^\infty} \leq \alpha_{R+1}\beta h^2$$

for all $n \leq N = m_{R+1}$, which ends the proof. ∎

Remark 4.6. For stable approximation operators (such as iecewise affine interpolations) we have $\|A\|_{L^\infty} = 1$ and the constant C in (4.10) depends linearly of the number of remappings. In any case, this number should not vary much with h when the latter is small, indeed the remapping frequency should essentially reflect the smoothness of the exact characteristic flow.

4.3. **Transport of smoothness**

In the above analysis we did not take advantage of the fact that the reconstructed flow was obtained with a smooth patching procedure (3.7). Nevertheless, this feature of the FBL method allows to estimate the smoothness of the transported solutions as time evolves, and this may yield enhanced error bounds in the remapped version of the scheme, since remapping errors strongly depend on the solution smoothness. We illustrate this property with the following result.

Theorem 4.7. *The density transported with the FBL scheme* (3.8) *of order* $r \in \{1, 2\}$ *satisfies*

$$\|f_h^{n,\mathrm{fbl}}\|_{W^{q,\infty}} \leq C\|f^0\|_{W^{q,\infty}}, \qquad 1 \leq q \leq r+1,$$

with a constant C *that depends on the exact flow* $F_{\mathrm{ex}}^{0,n}$ *and its inverse* $B_{\mathrm{ex}}^{0,n}$*, but not on* h*.*

Remark 4.8. With Lagrangian methods such as (1.6), stability estimates require strong constraints on the small parameters h and ε to avoid the well-known oscillation (the so-called particle noise) issue, see [30]. These constraints are demanding already for L^p stability as discussed in the introduction, and they are even stronger for higher order norms. In standard semi-Lagrangian methods where remappings are performed at each time step, the grid projection operators usually have stability constants higher than 1, which results in a priori constants of the form C^n at time t^n, with $C > 1$. With the FBL approach the situation is more favorable, since remappings are meant to be performed less frequently, as discussed in Remark 4.6.

Proof. We only give the proof in the case of the first order method ($r = 1$), as that of the second order is similar. By differentiating the transported density $f_h^{n,\mathrm{fbl}} = f^0(B_h^{0,n})$ one obtains

$$\partial_j f_h^{n,\mathrm{fbl}}(x) = \sum_{l=1}^{d}\partial_l f^0(B_h^{0,n}(x))\partial_j(B_h^{0,n})_l(x), \qquad 1 \leq j \leq d,$$

and

$$\partial_{j_2}\partial_{j_1} f_h^{n,\mathrm{fbl}}(x) = \sum_{l_1,l_2=1}^{d} \partial_{l_2}\partial_{l_1} f^0(B_h^{0,n}(x))\partial_{j_1}(B_h^{0,n})_{l_1}(x)\partial_{j_2}(B_h^{0,n})_{l_2}(x) + \sum_{l=1}^{d} \partial_l f^0(B_h^{0,n}(x))\partial_{j_2}\partial_{j_1}(B_h^{0,n})_l(x), \qquad 1 \le j_1, j_2 \le d.$$

This yields

$$|f_h^{n,\mathrm{fbl}}|_{W^{1,\infty}} \le C|B_h^{0,n}|_{W^{1,\infty}}|f^0|_{W^{1,\infty}}$$

and

$$|f_h^{n,\mathrm{fbl}}|_{W^{2,\infty}} \le C(|B_h^{0,n}|^2_{W^{1,\infty}}|f^0|_{W^{2,\infty}} + |B_h^{0,n}|_{W^{2,\infty}}|f^0|_{W^{1,\infty}})$$

with constants depending only on d. We are then left to estimate the smoothness of the reconstructed flow $B_h^{0,n} = \sum_{i\in\mathbb{Z}^d} B_{h,k^*(n,i)}^{0,n} S_{h,i}$, whose partial derivative reads

$$\partial_j(B_h^{0,n})_l = \sum_{i\in\mathbb{Z}^d} \Big(\partial_j\Big(B_{h,k^*(n,i)}^{0,n}\Big)_l S_{h,i} + \Big(B_{h,k^*(n,i)}^{0,n}\Big)_l \partial_j S_{h,i}\Big). \tag{4.12}$$

Here the first term is easily taken care of by using the fact that the Jacobian matrix of the local affine flow $B_{h,k}^{0,n}$ is the matrix D_k^n, see (3.4), for which an a priori bound is given in the Appendix, see (A.12). Bounding the second term is less obvious since an $1/h$ factor appears in the derivative of $S_{h,i}(x) = S((x-\xi)/h)$. To handle this term we then observe that the sum $\sum_{i\in\mathbb{Z}^d} \partial_j S_{h,i}$ vanishes, thanks to the unity partition property (3.6). Hence we can write, for all $x \in \mathbb{R}^d$,

$$\begin{aligned}\Big|\sum_{i\in\mathbb{Z}^d}\Big(B_{h,k^*(n,i)}^{0,n}\Big)_l \partial_j S_{h,i}\Big|(x) &= \Big|\sum_{i\in\mathbb{Z}^d}\Big(B_{h,k^*(n,i)}^{0,n} - B_{\mathrm{ex}}^{0,n}\Big)_l \partial_j S_{h,i}\Big|(x)\\ &\le \sum_{i\in\mathbb{Z}^d} \|B_{h,k^*(n,i)}^{0,n} - B_{\mathrm{ex}}^{0,n}\|_{L^\infty(\mathrm{supp}(S_{h,i}))}|\partial_j S_{h,i}|(x) \le Ch\end{aligned}$$

where we have used the local flow error estimate (4.5) and the bounded overlapping of the shapes $S_{h,i}$ (here the constant depends on the shape S and the flow $F_{\mathrm{ex}}^{0,n}$). This allows us to bound the first derivatives of $B_h^{0,n}$,

$$|B_h^{0,n}|_{W^{1,\infty}} \le C(F_{\mathrm{ex}}^{0,n}).$$

For the second derivatives we proceed similarly. Differentiating (4.12) and using the affine nature of the local flows, we write

$$\partial_{j_2}\partial_{j_1}(B_h^{0,n})_l = \sum_{i\in\mathbb{Z}^d} \Big(\partial_{j_1}\Big(B_{h,k^*(n,i)}^{0,n}\Big)_l \partial_{j_2} S_{h,i} + \partial_{j_2}\Big(B_{h,k^*(n,i)}^{0,n}\Big)_l \partial_{j_1} S_{h,i} + \Big(B_{h,k^*(n,i)}^{0,n}\Big)_l \partial_{j_2}\partial_{j_1} S_{h,i}\Big).$$

Again we can use the above trick and replace the local flows by local flow errors. In the last term the estimate (4.5) gives an h^2 factor that allows to take care of the $1/h^2$ term coming from the second derivatives of the scaled shape $S_{h,i}$, and in the derivatives we are lead to consider the local error on the backward Jacobian matrix, namely $D_{k^*(n,i)}^n - J_{B^{0,n}}(x)$ for $x \in \mathrm{supp}(S_{h,i})$, which is shown to be controlled by Ch by reasoning just as in the proof of Theorem 4.1. Thus, we finally have that

$$|B_h^{0,n}|_{W^{2,\infty}} \le C$$

which completes the proof for the first-order method. The case of the second-order FBL method is completely similar, using the fact that the local flow errors are estimated with one higher order of accuracy in the proof of Theorem 4.3. ∎

5. **Numerical results**

We now investigate the numerical efficiency of the proposed FBL method and compare it with several other methods, using either passive problems with given velocity fields or non-linear transport problems. Our comparisons involve a forward semi-Lagrangian (FSL) scheme [18, 17], a backward semi-Lagrangian (BSL) scheme [35, 34] and also the LTP and QTP methods described in Section 2. Here we do not show comparisons with fully Lagrangian methods involving smooth particle reconstructions (1.6) since the performances of LTP and QTP schemes have already been compared with such methods in previous works, see e.g. [8] for passive transport problems and [10] for Vlasov-Poisson problems where comparisons are done with a particle-in-cell (PIC) and high-order BSL schemes.

In all the numerical tests the particle shape functions are defined as two dimensional tensor products cubic B-splines $\varphi_h(x) = (\frac{1}{h})^2 \mathcal{B}_3(\frac{x_1}{h})\mathcal{B}_3(\frac{x_2}{h})$, with

$$\mathcal{B}_3(s) = \frac{1}{6}\begin{cases} (2-|s|)^3 & \text{if } 1 \le |s| < 2, \\ 4 - 6s^2 + 3|s|^3 & \text{if } 0 \le |s| < 1, \\ 0 & \text{elsewhere.} \end{cases}$$

The computational domain is the square $[0,1] \times [0,1]$ hence the number of particles N_p and the remapping grid size h are linked by $h^2 N_p = 1$.

5.1. **Passive transport problems**

As in [8] we consider several passive transport problems in 2d. The corresponding velocity fields are

- the reversible "swirling" velocity field proposed by LeVeque [26] to study the accuracy of high-resolution schemes for multidimensional advection problems,

$$u_{\rm SW}(t,x;T) := \cos\Big(\frac{\pi t}{T}\Big)\,\mathrm{curl}\,\phi_{\rm SW}(x) \quad \text{with} \quad \phi_{\rm SW}(x) := -\frac{\sin^2(\pi x_1)\sin^2(\pi x_2)}{\pi}$$

- another reversible velocity field emulating a Raylegh-Benard convection cell,

$$u_{\rm RB}(t,x;T) := \cos\Big(\frac{\pi t}{T}\Big)\,\mathrm{curl}\,\phi_{\rm RB}(x)$$

  with $\phi_{\rm RB}(x) := (x_1 - \frac{1}{2})(x_1 - x_1^2)(x_2 - x_2^2)$;

- and finally a constant non-linear rotation field derived from Example 2 in [7],

$$u_{\rm NLR}(x) := \alpha(x)\begin{pmatrix} \frac{1}{2} - x_2 \\ x_1 - \frac{1}{2} \end{pmatrix} \quad \text{with} \quad \alpha(x) := \Big(1 - \frac{\|x - (\frac{1}{2},\frac{1}{2})\|_2}{0.4}\Big)_+^3.$$

Here the form of $u_{\rm SW}$ and $u_{\rm RB}$ yields reversible problems: at $t = T/2$ the solutions reach a maximum stretching, and they revert to their initial value at $t = T$. As for the non-linear rotation field $u_{\rm NLR}$, it is associated with the exact backward flow

$$B^{0,n}(x) = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} + \begin{pmatrix} \cos(\alpha(x)t^n) & \sin(\alpha(x)t^n) \\ -\sin(\alpha(x)t^n) & \cos(\alpha(x)t^n) \end{pmatrix}\begin{pmatrix} x_1 - \frac{1}{2} \\ x_2 - \frac{1}{2} \end{pmatrix}, \tag{5.1}$$

and the exact solutions are given by $f(t^n,x) = f^0(B^{0,n}(x))$. In addition to the above velocity fields we consider the following initial data:

- smooth humps of approximate radius 0.2 given by

$$f^0_{\rm hump}(x;\bar{x}) := \frac{1}{2}\Big(1 + \mathrm{erf}\,\Big(\tfrac{1}{3}(11 - 100\|x - \bar{x}\|_2)\Big)\Big)$$

  and centered on $\bar{x} = (0.5, 0.4)$ or $(0.5, 0.7)$, depending on the cases ;

- and for the non-linear rotation field u_{NLR} we take an initial data corresponding to Example 2 from [7], i.e.,

$$f^0(x) := x_2 - \tfrac{1}{2}.$$

By combining the above values we obtain the three test-cases in Table 5.1, and accurate solutions are shown in Figures 5.1-5.3 for the purpose of illustration. In Table 5.1 we also give the respective time steps Δt used in the time integration of the particle trajectories. In every case indeed, the numerical flow F^n is computed with a RK4 scheme, and the time steps have been taken small enough to have no significant effect on the final accuracy. It happens that in every case we have $\Delta t = T/100$, but this is unintended.

Table 5.1. Definition of the benchmark test-cases

name	$u(t,x)$	$f^0(x)$	T	Δt
SW	$u_{\mathrm{SW}}(t,x;T)$	$f^0_{\mathrm{hump}}(x;\bar{x})$ with $\bar{x}=(0.5,0.7)$	5	0.05
RB	$u_{\mathrm{RB}}(t,x;T)$	$f^0_{\mathrm{hump}}(x;\bar{x})$ with $\bar{x}=(0.5,0.4)$	3	0.03
NLR	$u_{\mathrm{NLR}}(x)$	$x_2 - \frac{1}{2}$	50	0.5

From the convergence curves shown in Figures 5.1 to 5.4 we can draw the following observations:

- Overall, the two approaches (LTP/QTP and L/Q-FBL) reach similar accuracies, as there are no striking differences in the global behavior of the top and bottom curves in Figures 5.1 to 5.3.

- A look at the cpu numbers displayed in parenthesis, however, provides an important assessment: whereas the computational cost of the second order QTP scheme increases dramatically for growing remapping periods (which is caused by the stretching of the particle supports), for Q-FBL scheme it remains virtually equal to that of the first order methods (LTP and L-FBL). Thus the main objective of the new scheme is achieved.

- The theoretical gain of one convergence order of the FBL method noticed in Remark 4.2 is better seen in Figure 5.4 where the errors are measured in the less stringent L^2 norm. By increasing the remapping period one observes that the accuracy of the LTP simulations deteriorate from second to first order, whereas that of the L-FBL ones remain essentially of second order, which is consistent with our a priori estimate (4.1) for the transport error.

Remark 5.1 (on the convergence order)**.** As noticed above, the theoretical gain of one convergence order achieved by the FBL method compared to LTP/QTP approximations is hardly visible in Figures 5.1 to 5.3. Our explanation for this fact is that two errors are present in the convergence curves, namely transport errors and remapping errors. Because these curves show errors in L^∞, the steep gradients of the transported densities make the remapping errors dominate over most of the transport ones, and since all the methods shown here use the same remapping tool (cubic spline approximations), the remapping errors do not differ much between the various plots. We note however that when the remapping period increases the transport errors tend to become dominant (as there are less remappings and the transport approximations involve less regular characteristic flows), and the results tend to improve with the FBL method. This effect is clearly visible with the second-order runs (QTP vs. Q-FBL) in Figures 5.1 and 5.3, and this diagnosis is also confirmed by the L^2 convergence curves shown in Figure 5.4.

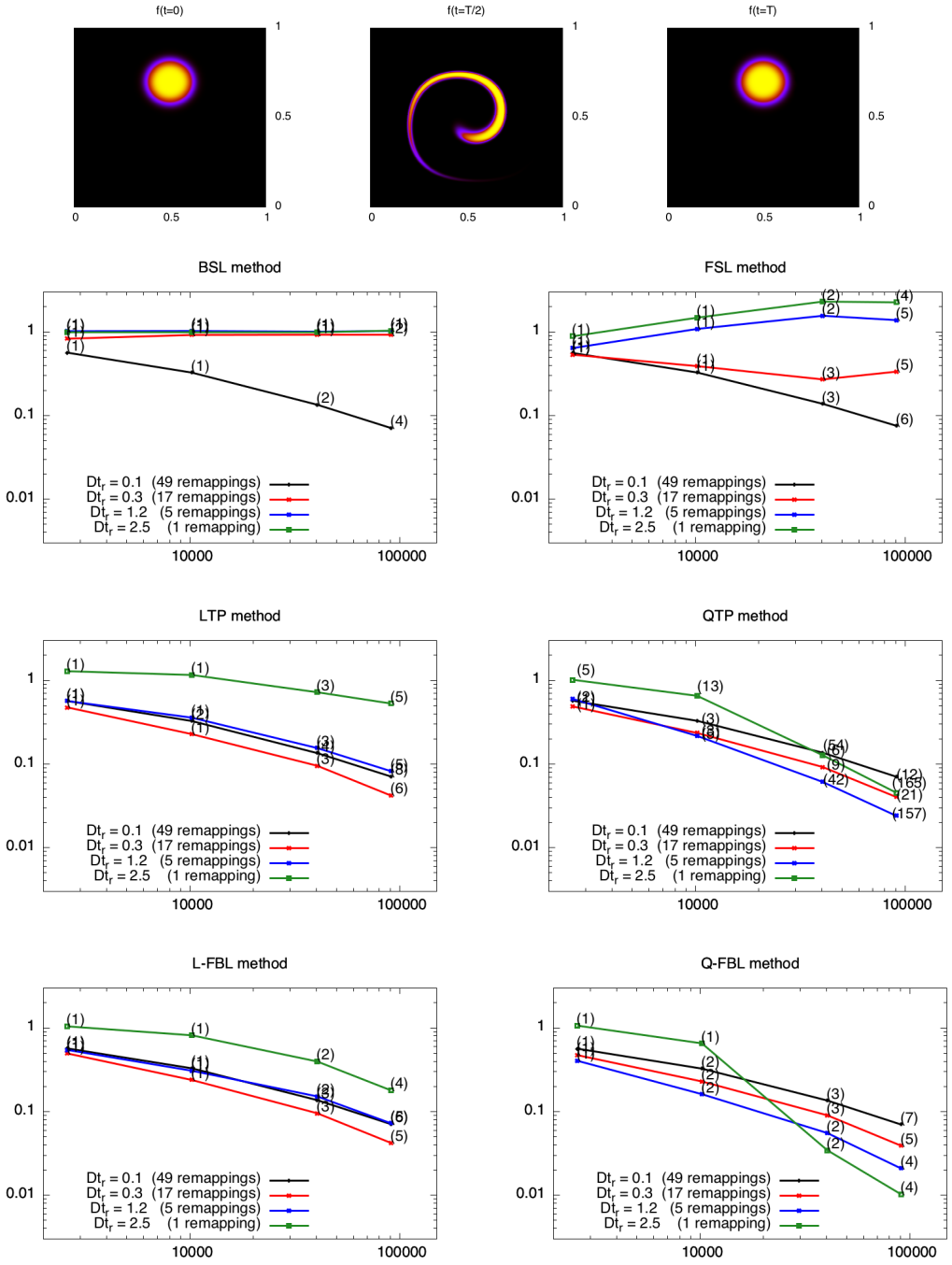


FIGURE 5.1. (Color) Convergence curves (relative L^∞ errors at $t = T$ vs. number of particles) for the reversible SW test case defined in Table 5.1, solved with different methods and different remapping periods Δt_r (for the BSL method which uses no particles, Δt_r is the standard time step). Numbers in parenthesis indicate the approximate cpu times (in seconds) for these runs. The first row shows the profile of the exact solution.

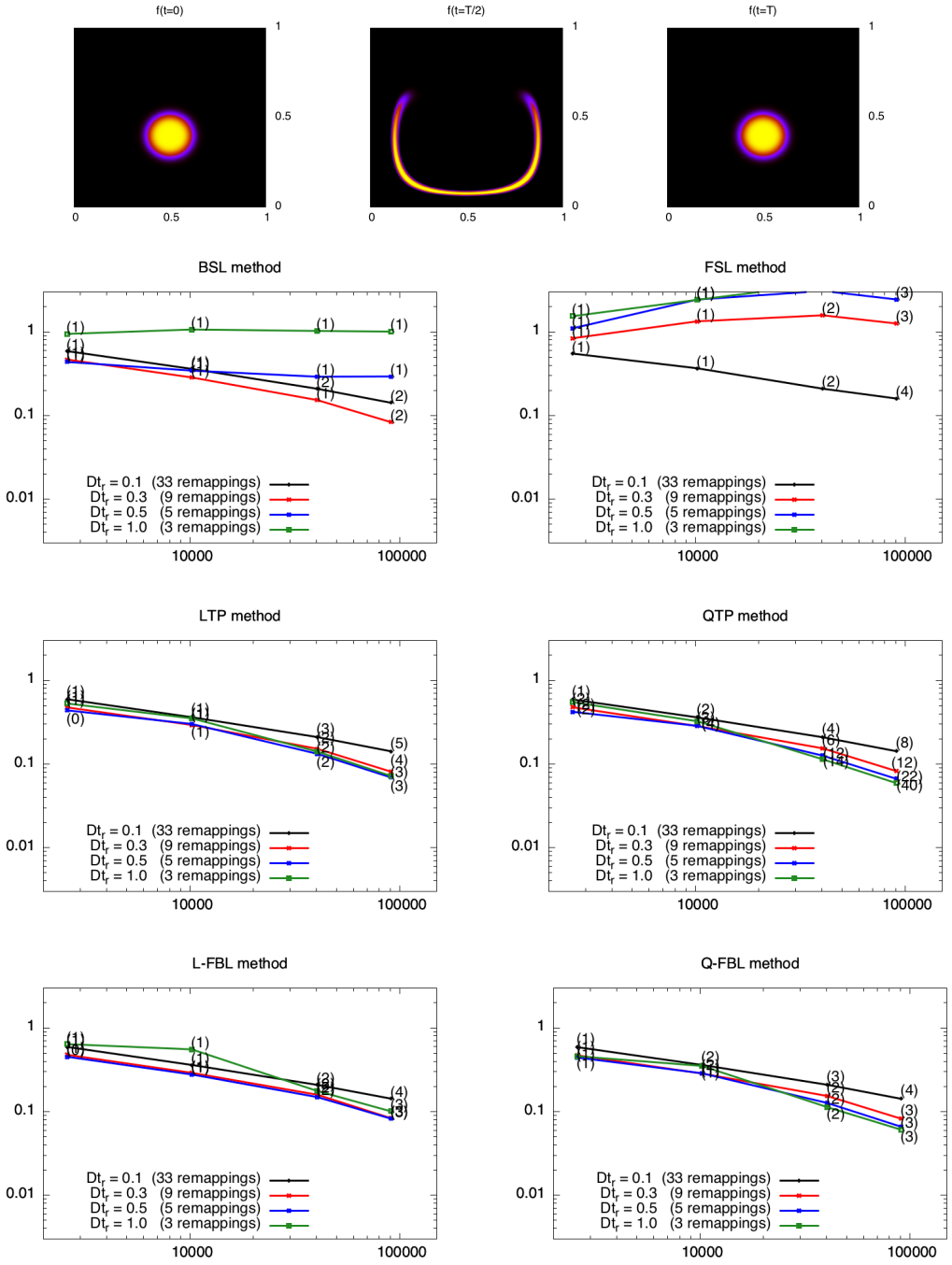


FIGURE 5.2. (Color) Convergence curves (relative L^∞ errors at $t = T$ vs. number of particles) for the reversible RB test case defined in Table 5.1, solved with different methods and different remapping periods Δt_r (for the BSL method which uses no particles, Δt_r is the standard time step). Numbers in parenthesis indicate the approximate cpu times (in seconds) for these runs. The first row shows the profile of the exact solution.

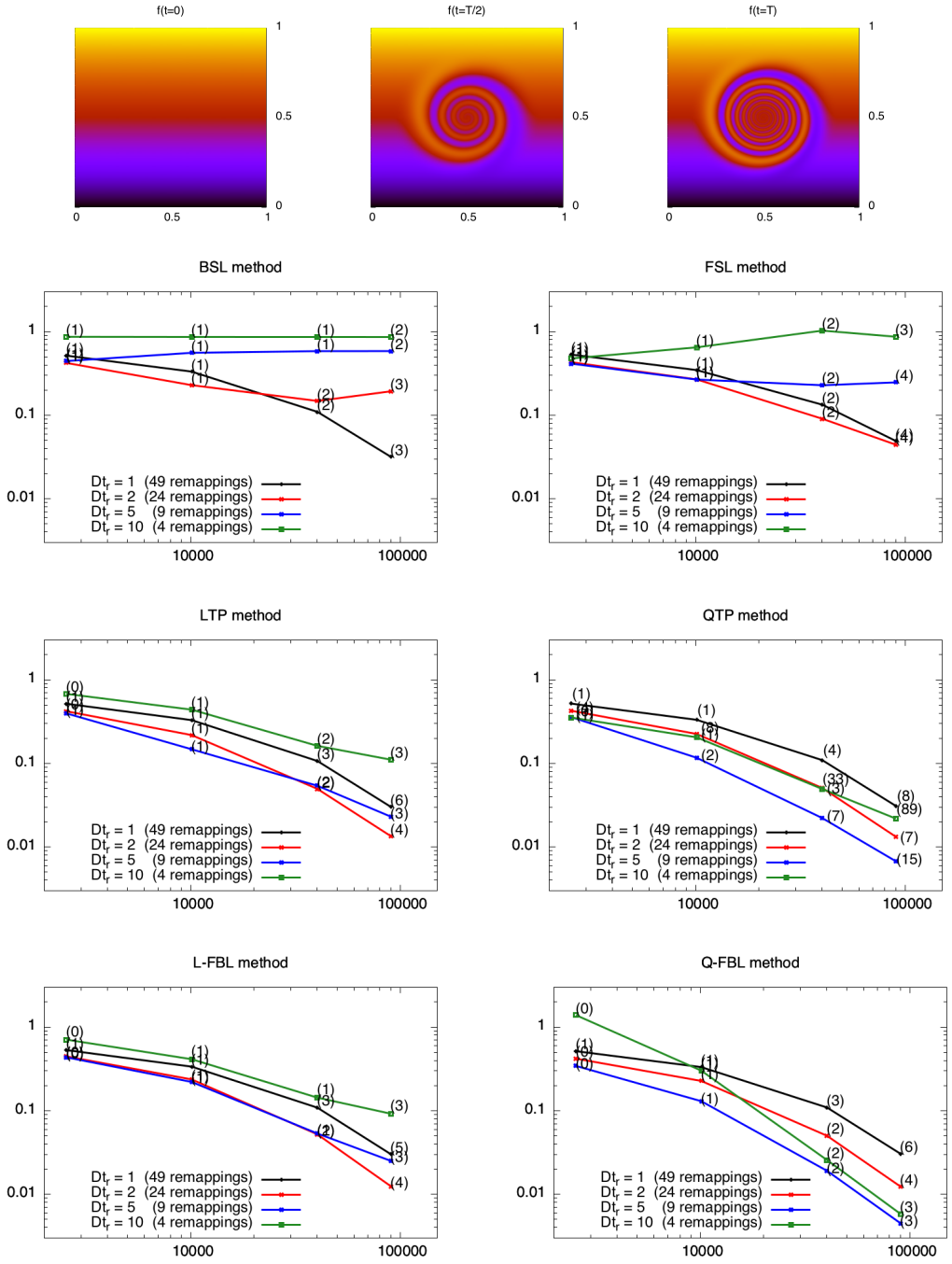


FIGURE 5.3. (Color) Convergence curves (relative L^∞ errors at $t = T$ vs. number of particles) for the NLR test case defined in Table 5.1, solved with different methods and different remapping periods Δt_r (for the BSL method which uses no particles, Δt_r is the standard time step). Numbers in parenthesis indicate the approximate cpu times (in seconds) for these runs. The first row shows the exact solution.

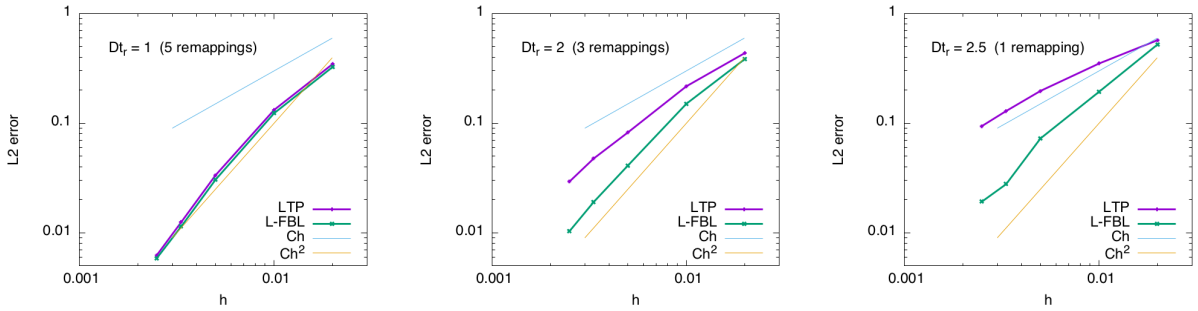


Figure 5.4. (Color) Convergence curves (relative L^2 errors at $t = T$ vs. h) for the reversible SW test case defined in Table 5.1, solved by the first order LTP and the L-FBL methods. Here the gain of one convergence order appears rather clearly when the remapping period is increased.

5.2. **Application to the 1D1V Vlasov-Poisson system**

In this section and the following one, we show some preliminary results obtained by applying our Forward-Backward Lagrangian approximation method to a couple of non-linear problems. As we aim at a minimal amount of modifications to existing particle codes, here we only use the FBL reconstruction formula (3.8) to re-initialize the particles with a given remapping frequency, in the spirit of (3.16). Specifically, we consider a 1D1V Vlasov-Poisson system

$$\begin{cases} \{\partial_t + v\partial_x + E(x,t)\partial_v\}\, f(t,x,v) = 0 \\ \partial_x E(t,x) = \displaystyle\int_{\mathbb{R}} f(t,x,v)\,\mathrm{d}v - n_b \end{cases} \qquad (x,v) \in \mathbb{R}^2, \tag{5.2}$$

which models the evolution of a normalized plasma of charged particles in a uniform neutralizing background cloud of density n_b.

To compute numerical approximations to the electric field E and the associated particle trajectories, we employ a standard particle-in-cell (PIC) method [22] which only sees point particles through the iecewise affine shape functions attached to the finite-difference 1D grid used for the field. On the remapping steps we then compute new particles using a cubic spline approximation to the FBL representation of the transported phase-space density f as in (3.11).

To assess our method we use the standard "weak" two-stream instability test-case [19, 31, 10] where the initial distribution reads

$$f^0(x,v) = \frac{2(1+5v^2)}{7\sqrt{2\pi}} e^{-\frac{v^2}{2}} \left(1 + A\left(\frac{\cos(2kx)+\cos(3kx)}{1.2} + \cos(kx)\right)\right)$$

with $k = \frac{1}{2}$ and a weak amplitude $A = 0.01$ for the perturbation. This test case is known to develop very thin filaments in the phase space that are difficult to resolve numerically.

In Figure 5.5 we compare the results of simulations where the particles have been pushed with a standard PIC method as described above, and remapped using either the first order LTP or the first order FBL reconstruction for the transported densities. The approximated densities are shown at $t = 53$ and the remapping period is always set to $\Delta t_R = 3$, which amounts to remapping every 15 time steps, which seemed to be a good value for virtually every run here. The simulations shown on the first line use a grid of 128×128 particles, whereas those on the second line use 512×512 particles. Again, the results show a similar accuracy for both methods, also similar to some high order semi-Lagrangian methods in the recent literature: here the 128×128 runs have small oscillations

compared to the right panel in Fig. 11 from Ref. [31] obtained with a conservative third order WENO BSL scheme using a 256×512 phase-space mesh, but our 512×512 run shows more features and has virtually no oscillations. As for the cpu times, FBL is less expensive than LTP (16 s. versus 10 s. for the high-resolution runs) due to the enhanced locality of the density reconstructions. We predict that in higher dimensions the gain will be much more significant.

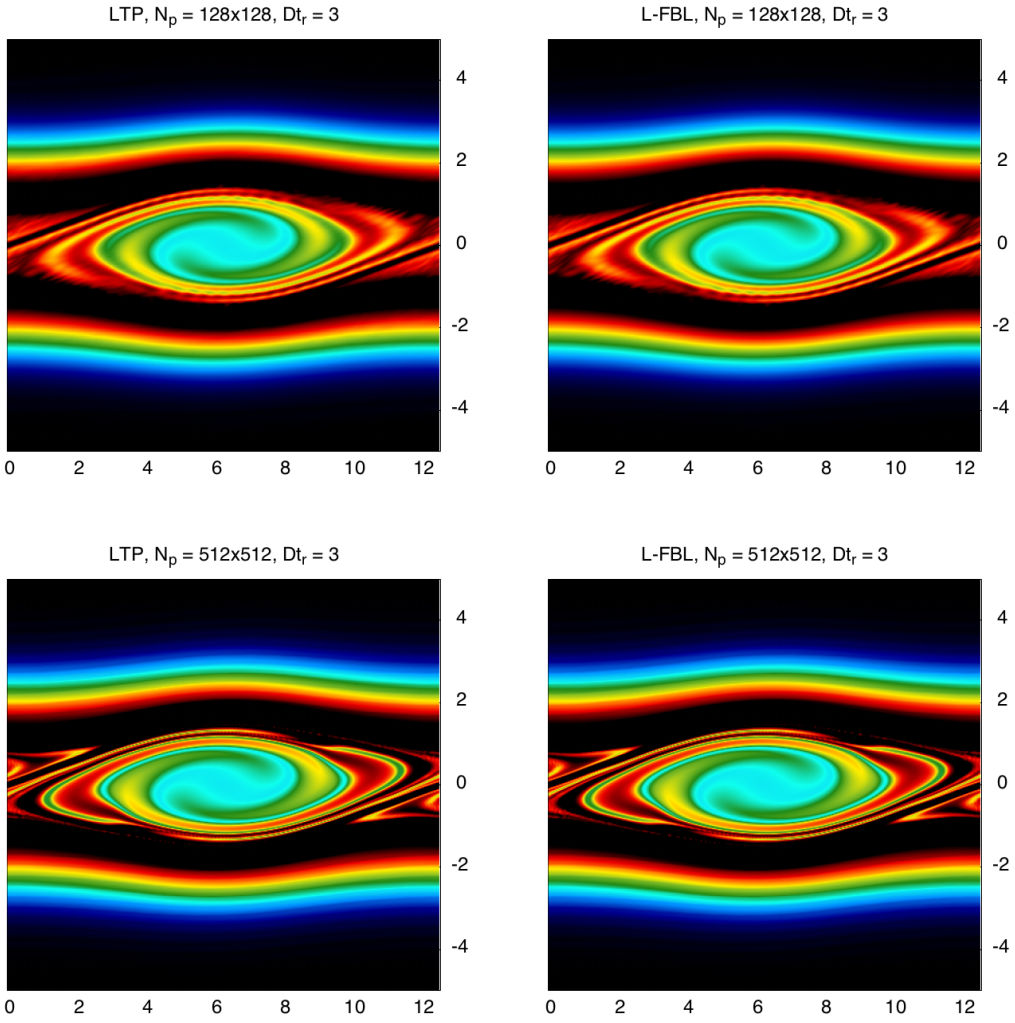


Figure 5.5. Vlasov-Poisson simulations of the two-stream instability described in Section 5.2. The approximated densities shown here have been computed up to $t = 53$ with a standard PIC scheme using 128×128 or 512×512 particles as indicated. On the left plots the particles are periodically remapped using an LTP reconstruction of the density, whereas on the right plots an L-FBL reconstruction is used. The approximated CPU times are close to 3 s. for the low-resolution runs (top row), 16 s. for the high resolution LTP run and 10 s. for the high resolution L-FBL run.

5.3. **Application to the 2D Euler equation**

We then consider the inviscid evolution of an elliptical vortex of compact support. The 2D incompressible Euler equations in vorticity-velocity form write:

$$\begin{aligned} \frac{\partial \omega}{\partial t} + u \cdot \nabla \omega &= 0, \\ \nabla \times u &= \omega, \\ \nabla \cdot u &= 0, \\ \lim_{|x| \to +\infty} |u| &= 0, \end{aligned} \tag{5.3}$$

and for the initial condition we take (like in [25] and [4])

$$\omega^0(x) = \omega_I^0(\sqrt{(x_1/0.8)^2 + (x_2/1.6)^2}), \qquad \omega_I^0(r) = 20(1 - \exp(-(2.56085/r)\exp(1/(r-1)))). \tag{5.4}$$

Here the vorticity ω plays the role of a transported density and is approximated by a particle representation. The velocity field can be related to ω using a Green's function formulation (see [16])

$$u = K * \omega \qquad \text{where} \qquad K(x) = \frac{1}{2\pi|x|^2}(-x_2, x_1). \tag{5.5}$$

This example has been used as a test case in [25] and in [4] to investigate Smooth Particle (Vortex) methods using adaptive and multilevel remapping techniques.

As we did for the Vlasov-Poisson system, we study here the combination of a standard method to push forward the particles centers and an accurate representation of the density (either with an LTP (2.4)-(2.3) or an FBL (3.8) reconstruction) to re-initialize the particles positions and weights at a given remapping frequency. Since a particle representation of the vorticity ω with Dirac masses leads to a velocity field that is singular on particles, it is classical to use a regularization $K_\varepsilon = K * \zeta_\varepsilon$ of the kernel K, with ζ_ε a smooth approximation of the Dirac mass. One can also consider an approximation of the vorticity with smoothed particles, that is

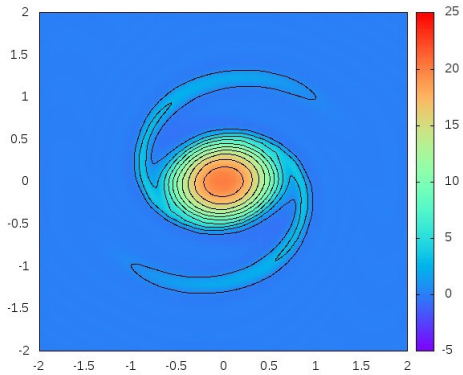
$$\omega_{h,\varepsilon}(t,x) = \sum_k w_k \zeta_\varepsilon(x - x_k(t)),$$

and use it directly in (5.5). In both cases, it leads to the regularized expression of the velocity of the k-th particle

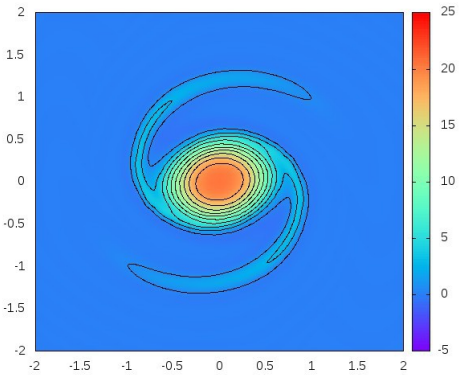
$$u_{h,\varepsilon}(t, x_k(t)) = \sum_j w_j K_\varepsilon(x_k(t) - x_j(t)). \tag{5.6}$$

Some examples of smoothing functions ζ_ε and resulting kernels K_ε can be found in [21] and [16].

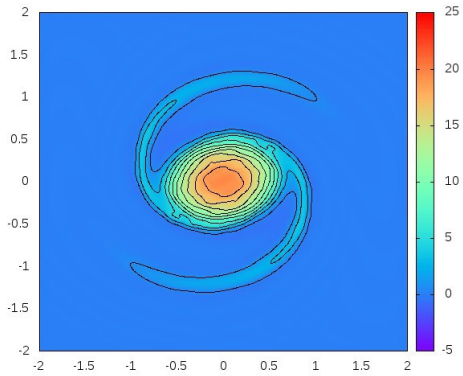
In Figure 5.6 we compare numerical vorticities obtained remapped with the first order LTP and FBL reconstructions. We see that the FBL method achieves an improved precision when the remapping period increases, which is reminiscent of the behavior already observed in Figure 5.4 for the Vlasov-Poisson simulations. Here the simulations involve grids of 50×50 particles corresponding to an average inter-particle distance of $h = 0.08$. In theses simulations the particle trajectories have been computed by applying an RK4 scheme with time step $\Delta t = 0.01$ to the above approximation (5.6) for the velocity, and for the smoothing of the kernel K we have set $\varepsilon = 0.01$. We note that this smoothing scale is much smaller than the inter-particle distance h, which is not sufficient to guarantee the convergence of the reconstructed vorticity in view of the classical analysis [21, 3, 32]. The qualitatively good results displayed in Figure 5.6 are thus a practical evidence of the beneficial influence of the accurate LTP and FBL reconstructions involved in the remappings.



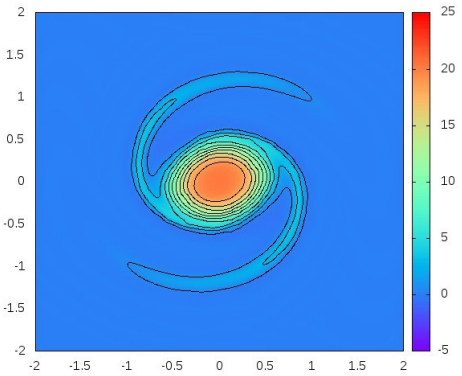
LTP method, $\Delta t_R = 0.1$



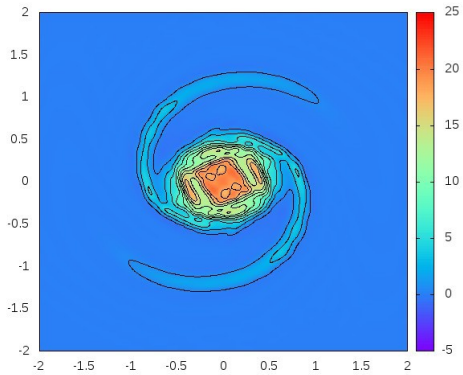
L-FBL method, $\Delta t_R = 0.1$



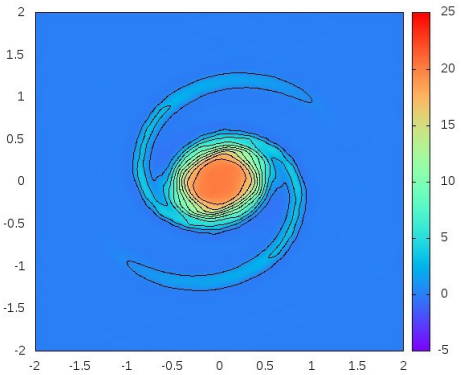
LTP method, $\Delta t_R = 0.15$



L-FBL method, $\Delta t_R = 0.15$



LTP method, $\Delta t_R = 0.2$



L-FBL method, $\Delta t_R = 0.2$

FIGURE 5.6. Vorticity contours for equation (5.3)-(5.4) with LTP or L-FBL scheme at time $t = 1.5$, using remapping periods Δt_R as indicated and with particles remapped on a 50×50 grid.

6. Conclusion

In this article we have introduced a novel method to represent the solution of a transport equation using an existing distribution of markers pushed forward. In standard smooth particle methods, the density is reconstructed as a sum of weighted shape functions centered on the markers positions. In the LTP and QTP methods recently developed by the authors, and in other similar methods proposed in the literature, the shapes of the particles are transformed according to a local expansion of the backward flow around each particle. These methods allow to obtain strong convergence properties, but at the price of stretching the particles supports which leads to a loss of locality.

In the Forward-Backward Lagrangian (FBL) method proposed here, locality is preserved by adopting a backward Lagrangian point of view to reconstruct the solution, which relies on an accurate representation of the initial density (or some reconstructed density at a the last remapping time) and on a global approximation of the backward flow. The latter is obtained by smoothly patching together local expansions of the flow with explicit formulas given in the appendix, around particles seen as (unweighted) flow markers.

Our a priori error analysis shows improved convergence rates for the resulting FBL reconstructions, compared with both standard smooth particles and LTP/QTP methods. Using first order (linear) flow expansions the L-FBL densities converge in $\mathcal{O}(h^2)$, and with second order (quadratic) expansions the Q-FBL densities converge in $\mathcal{O}(h^3)$, which represents a gain of one order compared to the LTP and QTP methods. Moreover, the smoothness of the FBL densities is stable in $W^{q,\infty}$ norms, with $q \leq 1$ for L-FBL and $q \leq 2$ for Q-FBL.

At the numerical level it is possible to observe the improved accuracy of the FBL reconstructions compared to the LTP ones, especially for increasing remapping periods. However, when remapping errors dominate our convergence studies tend to show that the overall quality of both methods is similar. From a CPU time point of view the LTP and L-FBL methods are quite comparable in two dimensions; but due to its good locality the Q-FBL method is much more efficient than the QTP one. Because of the stretching of particle supports the latter has indeed a large computational cost compared to first order methods when the remapping period increases, which has not been observed with the Q-FBL method. Based on this study we can advocate for quadratic FBL reconstructions and large remapping periods, at least when the underlying characteristic flow is smooth enough to justify a second order approximation. Both its implementation complexity and computational cost are close to those of the first order FBL method, and in several of the test-cases shown here the results are significantly better.

Additional results provided for non-linear problems such as the 1D1V Vlasov-Poisson system or the 2D Euler equation in vorticity form highlight two attractive features of the proposed approach compared to previous ones: on the one hand, its relative efficiency in terms of CPU time when the number of particles increases; on the other hand, its relative accuracy when the remapping period increases. These encouraging results call for more advanced comparisons with existing reconstruction methods. They will be the subject of future articles, in particular for higher dimensional Vlasov-Poisson codes currently under implementation.

Appendix A. Explicit approximation of the flow Jacobian from the particles

Using particles originally distributed on a Cartesian grid, i.e., $x_k^0 = hk$, $k \in \mathbb{Z}^d$, we compute the deformation matrices D_k^n approximating the Jacobian matrices of the backward flow at the particles positions (2.1), namely

$$J_{B_{\mathrm{ex}}^{0,n}}(x_k^n) = \left(\partial_j (B_{\mathrm{ex}}^{0,n})_i (x_k^n)\right)_{1 \leq i,j \leq d},$$

as follows. We first approximate the derivatives of the forward flow $F_{\rm ex}^{0,n}$ by finite differences involving the current particle positions $x_k^n = F_{\rm ex}^{0,n}(x_k^0)$. With a centered formula we define

$$J_k^n := \left(\frac{(x_{k+e_j}^n - x_{k-e_j}^n)_i}{2h} \right)_{1\leq i,j\leq d} \approx J_{F_{\rm ex}^{0,n}}(x_k^0)$$

and using the relation

$$J_{B_{\rm ex}^{0,n}}(x_k^n) J_{F_{\rm ex}^{0,n}}(x_k^0) = I_d \tag{A.1}$$

which follows by differentiating the identity $x = B_{\rm ex}^{0,n}(F_{\rm ex}^{0,n}(x))$ at x_k^0, we approximate $J_{B_{\rm ex}^{0,n}}(x_k^n)$ with

$$D_k^n := (J_k^n)^{-1}. \tag{A.2}$$

Since we consider a measure-preserving exact flow, we have $\det(J_{F_{\rm ex}^{0,n}}) = 1$ on $\mathbb{R}^d$ and it is reasonable to assume that the $d \times d$ matrix J_k^n is invertible. In the following Lemma we establish a sufficient condition for this, together with some a priori estimates for the resulting approximations.

Lemma A.1. *The approximate forward Jacobian matrix satisfies the a priori estimate*

$$\|J_k^n - J_{F_{\rm ex}^{0,n}}(x_k^0)\|_\infty \leq h^q \frac{|F_{\rm ex}^{0,n}|_{q+1}}{(q+1)!}, \quad q \in \{1,2\} \tag{A.3}$$

and the determinant error is bounded as

$$|\det(J_k^n) - 1| \leq \gamma^n(h) \quad \text{with} \quad \gamma^n(h) = 3d^2 |F_{\rm ex}^{0,n}|_1 \min_{q\in\{1,2\}} \Big(h^q \frac{|F_{\rm ex}^{0,n}|_{q+1}}{(q+1)!} \Big). \tag{A.4}$$

In particular, if h satisfies

$$h \leq h^*(F_{\rm ex}^{0,n}) := \max_{q\in\{1,2\}} \Big(\frac{3d^2}{2} |F_{\rm ex}^{0,n}|_1 \frac{|F_{\rm ex}^{0,n}|_{q+1}}{(q+1)!} \Big)^{-\frac{1}{q}} \tag{A.5}$$

then $\det(J_k^n) \geq \frac{1}{2}$ so that D_k^n is well defined, and we have the a priori estimate

$$\|D_k^n - J_{B_{\rm ex}^{0,n}}(x_k^n)\|_\infty \leq \min_{q\in\{1,2\}} \Big(h^q \frac{|F_{\rm ex}^{0,n}|_{q+1}}{(q+1)!} \Big) 2d^2 |F_{\rm ex}^{0,n}|_1^{2(d-1)}. \tag{A.6}$$

Proof. For conciseness, we denote in this proof

$$J_k^{n,\rm ex} = J_{F_{\rm ex}^{0,n}}(x_k^0) \quad \text{and} \quad D_k^{n,\rm ex} = J_{B_{\rm ex}^{0,n}}(x_k^n)$$

and using the semi-norms (1.8) we observe that

$$\max(\|J_k^{n,\rm ex}\|_\infty, \|J_k^n\|_\infty) \leq |F_{\rm ex}^{0,n}|_1. \tag{A.7}$$

Next we write two Taylor formulas for $s \mapsto F_{\rm ex}^{0,n}(x_k^0 + se_j)$ with $j = 1, \dots, d$, namely

$$F_{\rm ex}^{0,n}(x_k^0 + \sigma h e_j) = F_{\rm ex}^{0,n}(x_k^0) + \sigma h \partial_j F_{\rm ex}^{0,n}(x_k^0) + \int_0^{\sigma h} (\sigma h - s) \partial_j^2 F_{\rm ex}^{0,n}(x_k^0 + se_j)\, {\rm d}s, \quad \sigma = \pm 1 \tag{A.8}$$

and by taking their difference we obtain

$$(2h)^{-1} [F_{\rm ex}^{0,n}]_{x_k^0 - he_j}^{x_k^0 + he_j} = \partial_j F_{\rm ex}^{0,n}(x_k^0) + (2h)^{-1} \int_0^h (h-s)(\partial_j^2 F_{\rm ex}^{0,n}(x_k^0 + se_j) - \partial_j^2 F_{\rm ex}^{0,n}(x_k^0 - se_j))\, {\rm d}s.$$

This gives

$$\|J_k^n - J_k^{n,\rm ex}\|_\infty \leq h \frac{|F_{\rm ex}^{0,n}|_2}{2} \tag{A.9}$$

and also

$$\|J_k^n - J_k^{n,\rm ex}\|_\infty \leq h^2 \frac{|F_{\rm ex}^{0,n}|_3}{6} \tag{A.10}$$

which shows (A.3). Using next $\det(J_k^{n,\rm ex}) = 1$ and Lemma A.2, we find

$$|\det(J_k^n) - 1| \le d\|J_k^n - J_k^{n,\mathrm{ex}}\|_2(\|J_k^n - J_k^{n,\mathrm{ex}}\|_2 + \|J_k^{n,\mathrm{ex}}\|_2) \le 3d^2|F_{\mathrm{ex}}^{0,n}|_1 \min_{q\in\{1,2\}} \Big(h^q \frac{|F_{\mathrm{ex}}^{0,n}|_{q+1}}{(q+1)!}\Big) \quad \text{(A.11)}$$

since $\|M\|_2 \le \sqrt{d}\|M\|_\infty$ for $M \in \mathcal{M}_d(\mathbb{R})$. Here the upper bound corresponds to $\gamma^n(h)$, which shows (A.4). From now on we assume that h is as in (A.5), so that $\gamma^n(h) \le 1/2$, $\det(J_k^n) \ge 1/2$ and D_k^n is well defined. Using the formula $A^{-1} = \det(A)^{-1}C(A)^t$ involving the cofactor matrix $C(A)$ we then write, with $A = D_k^n = (J_k^n)^{-1}$,

$$\|D_k^n\|_\infty \le d\frac{\|J_k^n\|_\infty^{d-1}}{\det(J_k^n)} \le d(1-\gamma^n(h))^{-1}|F^{0,n}|_1^{d-1}. \quad \text{(A.12)}$$

With $A = D_k^{n,\mathrm{ex}} := (J_k^{n,\mathrm{ex}})^{-1}$, see (A.1), this also gives

$$\|D_k^{n,\mathrm{ex}}\|_\infty \le d\frac{\|J_k^{n,\mathrm{ex}}\|_\infty^{d-1}}{\det(J_k^{n,\mathrm{ex}})} \le d|F^{0,n}|_1^{d-1}. \quad \text{(A.13)}$$

In particular, writing $D_k^n - D_k^{n,\mathrm{ex}} = D_k^n(J_k^{n,\mathrm{ex}} - J_k^n)D_k^{n,\mathrm{ex}}$ gives

$$\|D_k^n - D_k^{n,\mathrm{ex}}\|_\infty \le \|D_k^n\|_\infty\|J_k^n - J_k^{n,\mathrm{ex}}\|_\infty\|D_k^{n,\mathrm{ex}}\|_\infty \le h\frac{d^2}{2}(1-\gamma^n(h))^{-1}|F^{0,n}|_1^{2(d-1)}|F^{0,n}|_2$$

using the second order estimate (A.9), while the third order estimate (A.10) leads to

$$\|D_k^n - D_k^{n,\mathrm{ex}}\|_\infty \le \|D_k^n\|_\infty\|J_k^n - J_k^{n,\mathrm{ex}}\|_\infty\|D_k^{n,\mathrm{ex}}\|_\infty \le h^2\frac{d^2}{6}(1-\gamma^n(h))^{-1}|F^{0,n}|_1^{2(d-1)}|F^{0,n}|_3.$$

This ends the proof. ∎

Lemma A.2. *For all A, $B \in \mathcal{M}_d(\mathbb{C})$ we have*

$$|\det(A) - \det(B)| \le d\big[\|A-B\|_2 + \|B\|_2\big]\|A-B\|_2$$

where $\|\cdot\|_2$ denotes the induced ℓ^2 norm for square matrices.

Proof. Let $\phi : t \mapsto tA + (1-t)B$. Using that the differential of the determinant is given by

$$D\det(M)H = \mathrm{Tr}(M^*H)$$

we can write

$$\begin{aligned}
|\det(A) - \det(B)| &= |\det(\phi(1)) - \det(\phi(0))| \\
&= \mathrm{Tr}\left(\phi(\theta)^*(A-B)\right), \qquad \text{with } \theta \in]0,1[\\
&= \theta\,\mathrm{Tr}((A-B)^*(A-B)) + \frac{1}{2}\mathrm{Tr}((A-B)^*B + B^*(A-B)) \\
&\le \theta\, d\varrho\left((A-B)^*(A-B)\right) + \frac{d}{2}\varrho\left((A-B)^*B + B^*(A-B)\right) \\
&\le \theta\, d\|A-B\|_2^2 + \frac{d}{2}\|(A-B)^*B + B^*(A-B)\|_2 \\
&\le\ d\|A-B\|_2^2 + d\|A-B\|_2\|B\|_2,
\end{aligned}$$

where $\varrho(M)$ is the spectral radius of a matrix M. ∎

Appendix B. Explicit approximation of the flow Hessian from the particles

To compute the quadratic deformation matrices $Q^n_{k,i}$ which approximate the Hessian matrices of the backward flow at the particles positions (2.6), namely

$$H_{(B^{0,n}_{\rm ex})_i}(x^n_k) = \big(\partial_{j_1}\partial_{j_2}(B^{0,n}_{\rm ex})_i(x^n_k)\big)_{1\le j_1,j_2\le d}, \qquad 1\le i\le d,$$

we follow the same principle as for the Jacobian matrices. First, using the current particles positions $x^n_k = F^{0,n}_{\rm ex}(x^0_k)$ we define approximate forward Hessian matrices as

$$H^n_{k,i} := \Big((h)^{-2}\sum_{\alpha_1,\alpha_2=0}^{1}(-1)^{\alpha_1+\alpha_2}\big(x^n_{k+\alpha_1 e_{j_1}+\alpha_2 e_{j_2}}\big)_i\Big)_{1\le j_1,j_2\le d} \approx H_{(F^{0,n}_{\rm ex})_i}(x^0_k) \tag{B.1}$$

which corresponds to finite differences on the original grid nodes $x^0_k = hk$, $k\in\mathbb{Z}^d$. Then, differentiating twice the identity $x = I(x) = B^{0,n}_{\rm ex}(F^{0,n}_{\rm ex}(x))$ we obtain

$$\left.\begin{aligned} 0 = \partial_{j_1}\partial_{j_2}(I)_i(x) = \sum_{l_1,l_2=1}^{d}\partial_{l_1}\partial_{l_2}(B^{0,n}_{\rm ex})_i(F^{0,n}_{\rm ex}(x))\partial_{j_1}(F^{0,n}_{\rm ex})_{l_1}(x)\partial_{j_2}(F^{0,n}_{\rm ex})_{l_2}(x)\\ +\sum_{l=1}^{d}\partial_l(B^{0,n}_{\rm ex})_i(F^{0,n}_{\rm ex}(x))\partial_{j_1}\partial_{j_2}(F^{0,n}_{\rm ex})_l(x)\end{aligned}\right\} \quad \begin{array}{c}\text{for}\\ 1\le i,j_1,j_2\le d.\end{array}$$

At $x = x^0_k$ and denoting for conciseness the exact Hessian matrices by

$$H^{n,\rm ex}_{k,i} = H_{(F^{0,n}_{\rm ex})_i}(x^0_k) \quad \text{ and } \quad Q^{n,\rm ex}_{k,i} = H_{(B^{0,n}_{\rm ex})_i}(x^n_k), \qquad 1\le i\le d, \tag{B.2}$$

this gives $0 = (J^{n,\rm ex}_k)^t Q^{n,\rm ex}_{k,i} J^{n,\rm ex}_k + \sum_{l=1}^d (D^{n,\rm ex}_k)_l H^{n,\rm ex}_{k,l}$, hence with (A.1),

$$Q^{n,\rm ex}_{k,i} = -(D^{n,\rm ex}_k)^t\Big(\sum_{l=1}^{d}(D^{n,\rm ex}_k)_{i,l}H^{n,\rm ex}_{k,l}\Big)D^{n,\rm ex}_k. \tag{B.3}$$

For the approximate backward Hessian matrix at x^n_k we thus set

$$Q^n_{k,i} := -(D^n_k)^t\Big(\sum_{l=1}^{d}(D^n_k)_{i,l}H^n_{k,l}\Big)D^n_k \tag{B.4}$$

where the approximate backward Jacobian matrix D^n_k is computed as in Appendix A.

Lemma B.1. *The approximate forward Hessian matrix defined in* (B.1) *satisfies the a priori estimate*

$$\|H^n_{k,i} - H_{(F^{0,n}_{\rm ex})_i}(x^0_k)\|_\infty \le C|F^{0,n}_{\rm ex}|_{W^{3,\infty}}h \tag{B.5}$$

with a constant C that depends only on the dimension d. Moreover if $h\le h^*(F^{0,n}_{\rm ex})$ as in (A.5)*, the backward ones $Q^n_{k,i}$ satisfy*

$$\|Q^n_{k,i} - H_{(B^{0,n}_{\rm ex})_i}(x^n_k)\|_\infty \le C_Q(F^{0,n}_{\rm ex})h \tag{B.6}$$

with a constant $C_Q(F^{0,n}_{\rm ex})$ that depends on $|F^{0,n}_{\rm ex}|_{W^{q,\infty}}$ with $1\le q\le 3$.

Proof. Expressing the finite differences in (B.1) as local averages of second derivatives of $F_{\rm ex}^{0,n}$, one easily verifies the estimate (B.5), as well as the bounds

$$\|H_{k,i}^n\|_\infty \leq C|F_{\rm ex}^{0,n}|_{W^{2,\infty}}, \qquad 1 \leq i \leq d \tag{B.7}$$

also satisfied by the exact $H_{k,i}^{n,\rm ex}$. To show (B.6) it then suffices to use the identities (B.3) and (B.4), together with the estimates (B.5) on H_k^n, (A.6) on D_k^n (with $q = 1$) and the bounds (A.12), (A.13), (B.7) satisfied by the exact and approximate (backward) Jacobian and (forward) Hessian matrices. ■

Appendix C. Main algorithms of the FBL method

We summarize here the algorithms that have been implemented to obtain the numerical results of Section 5. We denote:

- G_R the remapping grid, of step size h. Its nodes $x_k^0 = hk$, $k \in \mathbb{Z}^d$, correspond to the initial positions of the particles.

- G_F the flow grid on which the backward flow is reconstructed. In this paper we have chosen the same grid for G_F and G_R for simplicity, but the nodes of G_F are denoted ξ_i to avoid a confusion with the initial particle positions.

- G_V, an arbitrary grid on which the density $f_h^{n,\rm fbl}$ is reconstructed for visualization.

Algorithm 1 global FBL simulation

- **Initialization** Particles are created with positions set to the nodes $(x_k^0)_{k\in\mathbb{Z}^d}$ of the remapping grid G_R, and weights $(w_k^0)_{k\in\mathbb{Z}^d}$ computed from the initial density f^0.
- **Time evolution** For every time step n,
  - particles are pushed forward with a standard pusher (for example an RK4 method for the time scheme and a Particle-In-Cell method in the case of a non-linear transport problem),
  - if n is a visualization time step: $f_h^{n,\rm fbl}$ is reconstructed on the grid G_V,
  - if n is a remapping time step: particles positions are reset to $(x_k^0)_{k\in\mathbb{Z}^d}$ the nodes of G_R, and new weights $(w_k)_{k\in\mathbb{Z}^d}$ are computed from the reconstructed density $f_h^{n,\rm fbl}$. Previous particles positions and weights are discarded.

Algorithm 2 reconstruction of $f_h^{n,\mathrm{fbl}}$ at an arbitrary point x, given m the last remapping time step

- For every node ξ_i of the flow grid G_F,
  - the index of the closest particle is found[a]
    $$k^*(n,i) = \mathrm{argmin}_{k\in\mathbb{Z}^d} \|x_k^n - \xi_i\|_\infty$$
  - the backward Jacobian matrix $D^n_{k^*(n,i)}$ is computed and, in the case of second-order flows, the backward Hessian matrices $Q^n_{k^*(n,i)}$.
- At x, one then computes
  - $B_h^{m,n}(x)$ the approximate backward image of x, using the local flow matrices computed previously, see (3.14),
  - the approximate density value $f_h^{n,\mathrm{fbl}}(x) = f_h^m(B_h^{m,n}(x))$, following (3.13). This involves the density f_h^m on the Cartesian grid $G_R = (x_k^0)_{k\in\mathbb{Z}^d}$, i.e., the weights $(w_k^m)_{k\in\mathbb{Z}^d}$ computed at the last remapping time step m, see (3.11).

[a] We do not detail here the algorithm to find the closest particle of a grid node. For our analysis it is sufficient that the particle of index $k^*(n,i)$ satisfies $\max_{i\in\mathbb{Z}^d} \|x^n_{k^*(n,i)} - \xi_i\|_\infty \le Ch$ for some $C > 0$, see e.g. the proof of Theorem 4.1.

Acknowledgments

The authors thank Stéphane Colombi for fruitful discussions on the backward flow reconstructions, and Mehdi Badsi who implemented a 2d2v version of the LTPIC method showing the impact of the shape elongations on the performances. They also thank Antoine Le Hyaric who contributed to the implementation of a preliminary 2d2v version of the FBL method in the Selalib library, as well as the Selalib developers for their continued support. This work has been carried out within the framework of the Consortium and has received funding from the Euratom research and training program 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

[1] C. Alard and S. Colombi. A cloudy Vlasov solution. *Monthly Notices of the Royal Astronomical Society*, 359(1):123–163, May 2005.

[2] W.B. Bateson and D.W. Hewett. Grid and Particle Hydrodynamics. *Journal of Computational Physics*, 144:358–378, 1998.

[3] J.T. Beale and A. Majda. Vortex methods. II. Higher order accuracy in two and three dimensions. *Mathematics of Computation*, 39(159):29–52, 1982.

[4] M. Bergdorf, G.-H. Cottet, and P. Koumoutsakos. Multilevel adaptive particle methods for convection-diffusion equations. *Multiscale Modeling & Simulation*, 4(1):328–357, 2005.

[5] M. Bergdorf and P. Koumoutsakos. A Lagrangian particle-wavelet method. *Multiscale Modeling & Simulation*, 5(3):980–995, 2006.

[6] A. Biancalani, A. Bottino, S. Briguglio, A. Koenies, Ph. Lauber, A. Mishchenko, E. Poli, B.D. Scott, and F. Zonca. Linear gyrokinetic particle-in-cell simulations of Alfven instabilities in tokamaks. ⟨arXiv:1510.01945⟩, 2015.

[7] O. Bokanowski, J. Garcke, M. Griebel, and I. Klompmaker. An adaptive sparse grid semi-Lagrangian scheme for first order Hamilton-Jacobi Bellman equations. *Journal of Scientific Computing*, 55(3):575–605, 2013.

[8] M. Campos Pinto. Towards smooth particle methods without smoothing. *Journal of Scientific Computing*, 2014.

[9] M. Campos Pinto and F. Charles. Uniform Convergence of a Linearly Transformed Particle Method for the Vlasov–Poisson System. *SIAM Journal on Numerical Analysis*, 54(1):137–160, January 2016.

[10] M. Campos Pinto, E. Sonnendrücker, A. Friedman, D.P. Grote, and S.M. Lund. Noiseless Vlasov–Poisson simulations with linearly transformed particles. *Journal of Computational Physics*, 275(C):236–256, October 2014.

[11] Y. Chen, S.E. Parker, G. Rewoldt, S.-H. Ku, G.-Y. Park, and C.-S. Chang. Coarse-graining the electron distribution in turbulence simulations of tokamak plasmas. *Physics of Plasmas*, 15(5):055905, 2008.

[12] A. Cohen and B. Perthame. Optimal Approximations of Transport Equations by Particle and Pseudoparticle Methods. *SIAM Journal on Mathematical Analysis*, 32(3):616–636, October 2000.

[13] S. Colombi and C. Alard. A "metric" semi-Lagrangian Vlasov-Poisson solver. Submitted, 2016.

[14] C.J. Cotter, J. Frank, and S. Reich. The remapped particle-mesh semi-Lagrangian advection scheme. *Quarterly Journal of the Royal Meteorological Society*, 133(622):251–260, 2007.

[15] G.-H. Cottet, P. Koumoutsakos, and M.L.O. Salihi. Vortex Methods with Spatially Varying Cores. *Journal of Computational Physics*, 162(1):164–185, July 2000.

[16] G.H. Cottet and P. Koumoutsakos. *Vortex Methods: Theory and Practice*. Cambridge University Press, Cambridge, 2000.

[17] N. Crouseilles, T. Respaud, and E. Sonnendrücker. A forward semi-Lagrangian method for the numerical solution of the Vlasov equation. *Computer Physics Communications*, 180(10):1730–1745, October 2009.

[18] J. Denavit. Numerical Simulation of Plasmas with Periodic Smoothing in Phase Space. *Journal of Computational Physics*, 9:75–98, 1972.

[19] F. Filbet and E. Sonnendrücker. Comparison of Eulerian Vlasov solvers. *Computer Physics Communications*, 150:247–266, 2003.

[20] Ch. Gauger, P. Leinen, and H. Yserentant. The finite mass method. *SIAM J. Numer. Anal.*, 37(6):1768–1799, 2000.

[21] O.H. Hald. Convergence of Vortex Methods for Euler's Equations. II. *SIAM Journal on Numerical Analysis*, 16(5):726–755, October 1979.

[22] R.W. Hockney and J.W. Eastwood. *Computer simulation using particles*. Taylor & Francis, Inc, Bristol, PA, USA, 1988.

[23] T.Y. Hou. Convergence of a Variable Blob Vortex Method for the Euler and Navier-Stokes Equations. *SIAM Journal on Numerical Analysis*, 27(6):1387–1404, December 1990.

[24] P. Koumoutsakos. Inviscid Axisymmetrization of an Elliptical Vortex. *Journal of Computational Physics*, 138:821–857, December 1997.

[25] P. Koumoutsakos. Inviscid axisymmetrization of an elliptical vortex. *Journal of Computational Physics*, 138(2):821–857, 1997.

[26] R.J. LeVeque. High-resolution conservative algorithms for advection in incompressible flow. *SIAM Journal on Numerical Analysis*, pages 627–665, 1996.

[27] A. Magni and G.-H. Cottet. Accurate, non-oscillatory, remeshing schemes for particle methods. *Journal of Computational Physics*, 231(1):152–172, 2012.

[28] J.J. Monaghan. Extrapolating B. Splines for Interpolation. *Journal of Computational Physics*, 60:253, September 1985.

[29] R.D. Nair, J.S. Scroggs, and F.H.M. Semazzi. A forward-trajectory global semi-Lagrangian transport scheme. *Journal of Computational Physics*, 190(1):275–294, September 2003.

[30] W.M. Nevins, G.W. Hammett, A.M. Dimits, W. Dorland, and D.E. Shumaker. Discrete particle noise in particle-in-cell simulations of plasma microturbulence. *Physics of Plasmas*, 12:122305, 2005.

[31] J.-M. Qiu and A. Christlieb. A conservative high order semi-Lagrangian WENO method for the Vlasov equation. *Journal of Computational Physics*, 229:1130–1149, 2010.

[32] P.-A. Raviart. An analysis of particle methods. In *Numerical methods in fluid dynamics (Como, 1983)*, pages 243–324. Lecture Notes in Mathematics, Berlin, 1985.

[33] Selalib. Semi-Lagrangian Library. http://selalib.gforge.inria.fr/.

[34] E. Sonnendrücker, J. Roche, P. Bertrand, and A. Ghizzo. The semi-Lagrangian method for the numerical resolution of the Vlasov equation. *Journal of Computational Physics*, 149(2):201–220, 1999.

[35] A. Staniforth and J. Côté. Semi-Lagrangian integration schemes for atmospheric models-a review. *Monthly weather review*, 119:2206–2223, 1991.

[36] M. Unser and I. Daubechies. On the approximation power of convolution-based least squares versus interpolation. *Signal Processing, IEEE Transactions on*, 45(7):1697–1711, 1997.

[37] B. Wang, G.H. Miller, and P. Colella. A Particle-In-Cell method with adaptive phase-space remapping for kinetic plasmas. *SIAM Journal on Scientific Computing*, 33:3509–3537, 2011.