J. Fluid Mech. (2018), *vol.* 857, R2, doi:10.1017/jfm.2018.797

JFM RAPIDS
journals.cambridge.org/rapids

Machine learning the kinematics of spherical particles in fluid flows

Zhong Yi Wan[1] **and Themistoklis P. Sapsis**[1,†]

[1]Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

(Received 26 June 2018; revised 1 October 2018; accepted 1 October 2018)

Numerous efforts have been devoted to the derivation of equations describing the kinematics of finite-size spherical particles in arbitrary fluid flows. These approaches rely on asymptotic arguments to obtain a description of the particle motion in terms of a slow manifold. Here we present a novel approach that results in kinematic models with unprecedented accuracy compared with traditional methods. We apply a recently developed machine learning framework that relies on (i) an imperfect model, obtained through analytical arguments, and (ii) a long short-term memory recurrent neural network. The latter learns the mismatch between the analytical model and the exact velocity of the finite-size particle as a function of the fluid velocity that the particle has encountered along its trajectory. We show that training the model for one flow is sufficient to generate accurate predictions for any other arbitrary flow field. In particular, using as an exact model for trajectories of spherical particles, the Maxey–Riley equation, we first train the proposed machine learning framework using trajectories from a cellular flow. We are then able to accurately reproduce the trajectories of particles having the same inertial parameters for completely different fluid flows: the von Kármán vortex street as well as a two-dimensional turbulent fluid flow. For the second example we also demonstrate that the machine learned kinematic model successfully captures the spectrum of the particle velocity, as well as the extreme event statistics. The proposed scheme paves the way for machine learning kinematic models for bubbles and aerosols using high-fidelity DNS simulations and experiments.

Key words: bubble dynamics, computational methods, drops and bubbles

1. Introduction

Finite-size or inertial particle dynamics are often encountered in nature and technological applications either in the form of bubbles or aerosols. Their motion

---

† Email address for correspondence: sapsis@mit.edu

differs markedly from infinitesimal particle dynamics as they exhibit clustering and dispersive behaviours, which are absent for infinitesimal fluid elements. Several studies (Tang *et al.* 1992; Tio *et al.* 1993; Martin & Meiburg 1994; Vasiliev & Neishtadt 1994; Marcu, Meiburg & Newton 1995; Crowe, Troutt & Chung 1996) have analysed inertial particle dynamics in either analytically defined or numerically generated fluid flows. These studies are based on the Maxey–Riley (MR) equation (Maxey & Riley 1983), the equation of motion for small spherical particles in an unsteady non-uniform flow velocity field.

In Haller & Sapsis (2008) the authors performed a singular perturbation analysis of the MR equation and derived a slow manifold that governs the dynamics of sufficiently small particles. This low-dimensional manifold has the form of a modified velocity field, expressed in terms of the underlying flow velocity field and the particle parameters. The result is a reduced-order inertial equation that describes the particle kinematics. This analysis allows one to perform accurate time inversion, as the full MR equation becomes numerically unstable when integrated backwards due to its singular character. Moreover, due to the purely kinematic character, the inertial equation facilitates study of clustering properties (Sapsis & Haller 2010) and the computation of attractive manifolds for finite-size particles (see Sapsis & Haller 2009; Sapsis, Haller & Peng 2010; Sudharsan, Brunton & Riley 2016).

Despite the favourable properties of the inertial equation its use can be limited due to several reasons. First, the inertial equation is an asymptotic equation with respect to the particle size, and higher-order corrections require higher-order material derivatives of the flow field (Haller & Sapsis 2008), which can be hard to obtain reliably, especially for experimental data. Even for smaller particles, the MR equation may not be valid, depending on the characteristics of the flow. In fact, for the case of neutrally buoyant particles, Sapsis *et al.* (2011) revealed via a direct comparison with experiments that particles can have velocities different from those of the underlying flow field. Other studies (e.g. Daitche & Tél 2014) have suggested that the inclusion of the Basset–Boussinesq memory term can improve the performance of the MR equation, but in this case the derivation of a slow manifold approximation proves to be a formidable task.

Here we present a novel approach for the derivation of a kinematic equation for finite-size particles that relies on utilizing machine learning methods in combination with a basic physical model for the particle motion, in this case the standard MR equation. We employ the blended machine learning framework that was formulated in Wan *et al.* (2018). The basic idea of this method is to machine learn the mismatch between data and an imperfect model that captures some of the underlying physics of the problem. Learning is performed using a long short-term memory (LSTM) recurrent neural network (RNN). The choice of an LSTM-RNN is important as it naturally incorporates memory into the resulting model. As a basic model for the underlying physics we use the first-order approximation of the slow manifold that is associated with the standard MR equation.

Note that while a purely data-driven approach using LSTM-RNN (e.g. Vlachas *et al.* 2018) could have been employed, here we choose to maintain as much as possible of the underlying basic physics through the derived inertial equation from the MR model. This strategy has certain advantages, primarily related to the fact that the mismatch between the two models has much smaller variance and can be learned more efficiently. On the other hand learning the full model would require more data points and would ignore the MR equation, which may not be perfect but still captures some important aspects of the particle motion.

Therefore, in the present context, the combination of a basic physical model and a machine learning framework has the following favourable properties: (i) the resulting model is not limited by the particle size, as is the case in the asymptotic model – at least to the extent that finite-size particles are small enough that they can be described as points, (ii) we can directly employ data from experiments or high-fidelity DNS models, (iii) the model utilizes all the available physics for regions where data are not available, (iv) we have direct modelling of memory effects. We present our results for the case of spherical bubbles, but the extension of the analysis for aerosols is straightforward. Specifically, we train the blended model using data from the full (non-reduced) MR equation for a simple cellular flow. We then assess the learned model for different flows, a flow behind a cylinder and a two-dimensional turbulent flow, and compare the performance with the inertial equation. We perform this comparison both at the trajectory and the statistics level.

2. Kinematics of spherical particles

Let $\boldsymbol{u}(\boldsymbol{x}, t)$ be the velocity field of a two-dimensional incompressible flow field of density ρ_f. We denote by $\boldsymbol{x}(t)$ the path of a finite-size particle of density ρ_p immersed in a fluid field with characteristic length L. For spherical particles of radius $a \ll L$, the Lagrangian particle velocity $\boldsymbol{v}(t)$ is assumed to satisfy the Maxey–Riley (MR) equation of motion (Maxey & Riley 1983; Haller & Sapsis 2008):

$$\dot{\boldsymbol{x}} = \boldsymbol{v}, \tag{2.1}$$

$$\dot{\boldsymbol{v}} = -\frac{1}{\epsilon}(\boldsymbol{v} - \boldsymbol{u}) + \frac{3R}{2}\frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t}, \tag{2.2}$$

where $\mathrm{D}/\mathrm{D}t \equiv \partial/\partial t + \boldsymbol{u}\boldsymbol{\cdot}\nabla$ denotes the material derivative and $\epsilon = 1/\mu \ll 1$ with inertial parameter μ defined as

$$\mu = \frac{R}{St}, \quad R = \frac{2\rho_f}{\rho_f + 2\rho_p}, \quad St = \frac{2}{9}\left(\frac{a}{L}\right)^2 Re. \tag{2.3a–c}$$

Here Re and St denotes the Reynolds and particle Stokes number respectively. The density ratio R distinguishes neutrally buoyant particles ($R = 2/3$) from aerosols ($R < 2/3$) and bubbles ($2/3 < R < 2$). Note that the larger the inertia parameter μ, the less significant the effect of inertia; in the $\mu \rightarrow \infty$ limit, this equation describes the motion of a passive ideal tracer particle.

2.1. *Slow manifold approach*

For all $\epsilon > 0$, the MR equation (2.2) admits an invariant slow manifold (Haller & Sapsis 2008) which has the asymptotic form

$$M_\epsilon = \left\{(\boldsymbol{x}, \boldsymbol{v}) : \boldsymbol{v} \equiv \tilde{\boldsymbol{u}}(\boldsymbol{x}, t) = \boldsymbol{u}(\boldsymbol{x}, t) + \epsilon\left[\frac{3R}{2} - 1\right]\frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t} + O(\epsilon^2)\right\}. \tag{2.4}$$

Moreover, as shown in Sapsis & Haller (2008), although M_ϵ exists for all values of ϵ, it globally attracts all trajectories, in $O(\epsilon)$ time, if and only if the following condition is satisfied:

$$\lambda_{min}[\boldsymbol{S}(\boldsymbol{x}, t) + \mu \boldsymbol{I}] > 0, \quad \text{where } \boldsymbol{S}(\boldsymbol{x}, t) = \frac{\nabla\tilde{\boldsymbol{u}}_\epsilon(\boldsymbol{x}, t) + \nabla\tilde{\boldsymbol{u}}_\epsilon(\boldsymbol{x}, t)^{\mathrm{T}}}{2}. \tag{2.5}$$

Here it is important to note that for a fixed ϵ, whether M_ϵ is globally attracting or not depends entirely on properties of the flow local to the particle and is unaffected by the velocity of the particle itself. Correspondingly, when a particle passes by a region where condition (2.5) is violated, its motion would diverge from M_ϵ, which repels all close-by trajectories.

Therefore, under the premise that (2.5) is satisfied along a particle trajectory and $M(t)$ is globally attracting for all $t > 0$, (2.4) provides an alternative description of the particle kinematics at all times. Keeping the leading-order terms, we obtain the inertial equation:

$$\dot{\boldsymbol{x}}(t) \equiv \boldsymbol{v}^*(t) = \boldsymbol{u}(\boldsymbol{x}, t) + \epsilon \left[\frac{3R}{2} - 1\right] \frac{\mathrm{D}\boldsymbol{u}}{\mathrm{D}t} + O(\epsilon^2). \tag{2.6}$$

Compared to the full MR equation (2.2), the inertial equation offers great modelling advantages as it has only half the dimension but non-zero divergence, which allows it to capture the clustering and dispersion of non-neutrally buoyant particles arising from finite-size effects. However, the accuracy of this approximation will be significantly compromised if (i) ϵ becomes larger and (ii) the local flow field is fast-changing either in space or time ($\boldsymbol{u}(\boldsymbol{x}, t)$ has large partial derivatives). When either condition is satisfied, higher-order effects become non-negligible. Although closed-form analytical expressions for these high-order terms are available, they involve calculating high-order material derivatives which are difficult to obtain for arbitrary flow fields. Hence, we plan to utilize the first approximation of the inertial equation and combine this with available data in order to machine learn the higher-order corrections.

3. Blended slow manifold model

We aim to develop a new kinematic model blending the leading-order slow manifold approximation with a machine learning (ML) complement. The main purpose of the ML component is to identify when higher-order effects become important based on the local flow states encountered by the particle, and provide an accurate numerical estimate for the resulting kinematics. At the same time, this additional component does not raise the system dimensionality and keeps the model computationally efficient. More specifically, our proposed model follows the blended strategy in Wan *et al.* (2018) and has the form

$$\dot{\boldsymbol{x}}(t) = \underbrace{\boldsymbol{u}(\boldsymbol{x}, t) + \epsilon \left(\frac{3R}{2} - 1\right) \frac{\mathrm{D}\boldsymbol{u}(\boldsymbol{x}, t)}{\mathrm{D}t}}_{\text{leading-order approx. } \boldsymbol{v}^*(t)} + \underbrace{G(\boldsymbol{\xi}(t), \boldsymbol{\xi}(t-\tau), \boldsymbol{\xi}(t-2\tau), \ldots)}_{\text{data-driven model } \boldsymbol{v}_d(t)}, \tag{3.1}$$

which is a direct sum of the inertial model kinematics $\boldsymbol{v}^*$ and output of a data-driven model $\boldsymbol{v}_d$. Here τ is a time lag parameter and $\boldsymbol{\xi}$ is a four-dimensional vector of flow states experienced by the particle along its own path at time t:

$$\boldsymbol{\xi}(t) = \left[\boldsymbol{u}(\boldsymbol{x}, t), \frac{\mathrm{D}\boldsymbol{u}(\boldsymbol{x}, t)}{\mathrm{D}t}\right]. \tag{3.2}$$

These flow states are closely associated with the force exerted on the particle by the surrounding fluid. With the proposed model form, we aim to deduce the velocity of the particle from the forces it perceives by learning a data-driven function

$G(\cdot)$ that maps the time series in $\boldsymbol{\xi}$ to the corresponding series in $\boldsymbol{v}_d$ – velocity difference between the particle and the leading-order inertial model. Time history/delay coordinates are included in the model to compensate for the fact that $\boldsymbol{\xi}(t)$ alone does not contain the full information required to evaluate the velocity on the slow manifold at time t. Note that the choice for the components of $\boldsymbol{\xi}$ is by no means unique and will likely affect the number of delays required for an accurate G mapping.

We utilize RNNs as our learning method for G. RNNs are a special type of neural network that provides a sequence-to-sequence mapping. The network reads input one at a time and computes the output from a running hidden state that is updated at every step based on the previous hidden state and the new input values, thus respecting the order of the input sequence. In particular, we utilize the LSTM (Hochreiter & Schmidhueber 1997), a special RNN unit, to map between discrete-time $\boldsymbol{\xi}$ and $\boldsymbol{v}_d$ sequences. We use one LSTM layer with 128 hidden units, followed by one dense/fully connected layer with 64 hidden units and a rectified linear unit (ReLU) activation function to connect the input and output. The resulting model has a total of 76 032 degrees of freedom (DOFs).

The model DOFs are optimized with respect to the mean squared error loss on the output $\boldsymbol{v}_d$. For our studies, training data are generated by directly integrating the original MR equation (2.2) and recording the model-required quantities (particle, flow velocities and material derivatives) along the path at some small uniform time step τ. We treat these trajectories as the ground truth from which the model learns to predict the discrepancy between particle velocity and the leading-order manifold approximation. Predictions made in this way are single step because model output is not used to update the position of the particle and thus does not impact the model inputs in the future, regardless of any error made in the process.

In a more practical prediction scenario, we often wish to forecast the trajectory of a particle for more than a single step. In this case, the predicted velocity fully dictates the particle position at the next time instant, which in turn determines the next model input. As a result, the model interacts with a flow database and queries for the next input based on the predicted particle position. Furthermore, an initial spin-up input sequence (typically of a few steps only) is needed such that zero initializations to the hidden states do not compromise performance. Figure 1 shows the overall multistep prediction algorithm with first-order time integration and the corresponding flow chart. We combine two such model instances, staggered in time, to achieve fourth-order integration accuracy. Finally, although it is possible to train the model by minimizing its multistep error (see Wan *et al.* 2018), we do not consider this option, mainly because the model would otherwise start to 'anticipate' future flow state based on training data, which is undesirable as we generalize the model to predict in time-dependent flows not encountered during training.

4. Applications

We demonstrate the proposed modelling methodology using three different flow problems. The first example consists of a simple two-dimensional flow with a simple periodic structure. We assume access to an excess amount of data (more than sufficient to accurately establish the functional approximation) and train the blended model to illustrate the main ideas of the proposed method. The second and third examples involve more complex flow structures – flow around a cylinder and turbulence in a periodic domain, respectively. We directly apply the models learned in the first example with no additional training and showcase the generalization capabilities of the proposed method to more realistic scenarios.

(*a*)

```
Input: time step $\tau$, number of set-up steps $m$,
       start time $t_0$, spin-up input sequence
       $\Xi = \{\boldsymbol{\xi}(t_0 - k\tau)\}_{k=0}^{m}$, start position $\boldsymbol{x}_0$
Initialization:
    $t \leftarrow t_0, \boldsymbol{x} \leftarrow \boldsymbol{x}_0$;
    run set-up sequence $\Xi$ through model;
    $\boldsymbol{h} \leftarrow$ final model hidden states, $\boldsymbol{v}_d \leftarrow$ final
    $\boldsymbol{v}^* \leftarrow$ lead-order approximation at $t_0$;
    $\boldsymbol{v} \leftarrow \boldsymbol{v}_d + \boldsymbol{v}^*$
repeat
    $t \leftarrow t + \tau$;
    $\boldsymbol{x} \leftarrow \boldsymbol{x} + \tau\boldsymbol{v}$;
    $\boldsymbol{\xi} \leftarrow$ query flow data at $\boldsymbol{x}$ and t;
    $\boldsymbol{v}_d \leftarrow$ evaluate based on $\boldsymbol{h}$ and $\boldsymbol{\xi}$;
    $\boldsymbol{h} \leftarrow$ update to newest model hidden states;
    $\boldsymbol{v}^* \leftarrow$ lead-order approximation at $t$;
    $\boldsymbol{v} \leftarrow \boldsymbol{v}_d + \boldsymbol{v}^*$
until last step of prediction;
```

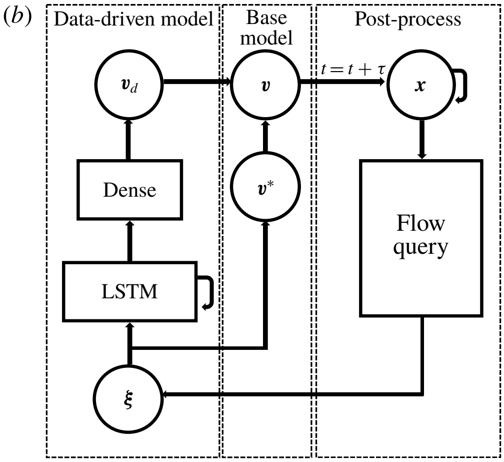

FIGURE 1. (*a*) Multistep trajectory prediction algorithm with first-order accurate time integration and (*b*) corresponding flow chart.

4.1. *Cellular flow*

We first consider the two-dimensional cellular flow model (Babiano *et al.* 2000; Cartwright *et al.* 2010) representing a lattice of oscillating vortices or roll cells. The flow is defined by the stream function

$$\psi(x, y, t) = A\cos(x + B\sin(\omega t))\cos(y) \tag{4.1}$$

in the $x, y \in [-\pi/2, \pi/2]$ domain with periodic boundary conditions. The fluid velocity field can be derived by letting $\boldsymbol{u} = (-\psi_y, \psi_x)$. Note that the time dependence in this flow can be suppressed by setting $B = 0$. Parameter A determines the strength of the vortices involved. Figure 2(*a*) shows the instantaneous velocity field at $t = 0$ and contours of λ_{min}, the minimum eigenvalue defined in (2.5). Figure 2(*b*) shows the time series for the velocity difference between the particle and flow velocities, as well as between the particle and the leading-order approximation, of a bubble ($\epsilon = 0.01$, $R = 1.55$) in a cellular flow with $A = 47$, $B = 4.4$ and $\omega = 11.5$. These flow parameters result in large spatial and temporal derivatives in the velocity field, which leads to a large difference between the particle and flow velocities despite a small ϵ. In consequence, the effects of higher-order terms become prominent.

We apply the proposed method to learn a hybrid model for the slow manifold at fixed particle parameters $\epsilon = 0.01$ and $R = 1.55$. Data are simulated by integrating (2.2) in cellular flow prescribed by (4.1) with random parameters drawn from uniform distributions: $A \sim \mathcal{U}[30, 50]$, $B \sim \mathcal{U}[0, 5]$ and $\omega \sim \mathcal{U}[0, 4\pi]$. The randomization serves to inject more variety into the flow patterns seen by the model during training, and thus improve generalization. 1000 trajectories are generated at 1500 time steps of 0.002. 80% of the data are used for training and the remainder is divided equally for validation and testing. After 300 epochs, the model is able to achieve a single-step mean squared error (MSE) of 3×10^{-3}, averaged over all time steps and validation cases. This is almost perfect prediction, especially considering the fact that the leading-order approximation $\boldsymbol{v}^*$ has a MSE of 14.61 and the mean $L2$-norm of $\boldsymbol{v} - \boldsymbol{u}$ is 25.18.

In figure 2(*c*–*e*) we compare the multistep prediction skill of the proposed RNN-blended model with the leading-order approximation $\boldsymbol{v}^*$ through an example test case.

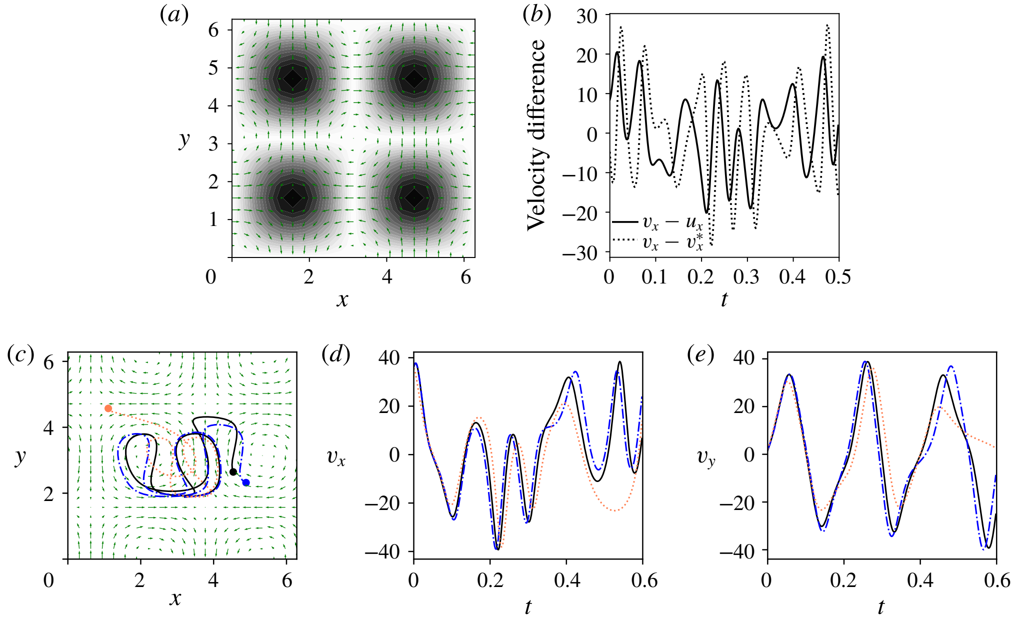

FIGURE 2. Cellular flow. (*a*) Velocity field at $t=0$ and magnitude of λ_{min} (background shading: lighter is higher λ_{min}). (*b*) Velocity difference between the particle and flow, and between the particle and leading-order approximation against time for a sample trajectory. (*c*–*e*) 300-step predictions made using the blended slow manifold model (— · —) and the leading-order approximation alone (······). (*c*) Trajectory in (x, y) space (ending flow field shown in the background; solid circles mark the ending positions). (*d*,*e*) Predicted velocity time series in the x and y directions, respectively. Truth (——) from integration of (2.2) is plotted for comparison. Flow parameters are $A=32$, $B=3.2$ and $\omega=9.2$. Plotted trajectories start from $(x, y)=(3.20, 1.88)$ at $t=0$.

Due to much smaller single-step errors, the trajectory predicted by the blended model is able to remain close to the truth for a significant amount of time, despite being in a rapidly changing and unstable flow environment.

4.2. *von Kármán vortex street*

In a second example we consider particle motion in the von Kármán vortex street model first studied in Jung, Tél & Ziemniak (1993). Finite-size particle motion in this flow has been studied in Benczik, Toroczkai & Tél (2002) and Haller & Sapsis (2008), who showed the existence of attractors for certain parameter values. The flow is governed by a stream function, and for simulation the system parameters are chosen such that the model approximates the Navier–Stokes solution for this geometry at $Re \approx 250$ with a flow period of $T_c = 1.107$.

We apply the model learned from the cellular flow trajectories in the previous example directly to the von Kármán vortex flow, since the learned slow manifold dynamics should be applicable regardless of the flow field encountered. In practice, we have found that this is indeed the case, demonstrating the robustness of the proposed approach. We have found that it is important to ensure that the model is never asked to extrapolate significantly, both in terms of the input and target. A rough way to check is to plot and compare the marginal density of each input and

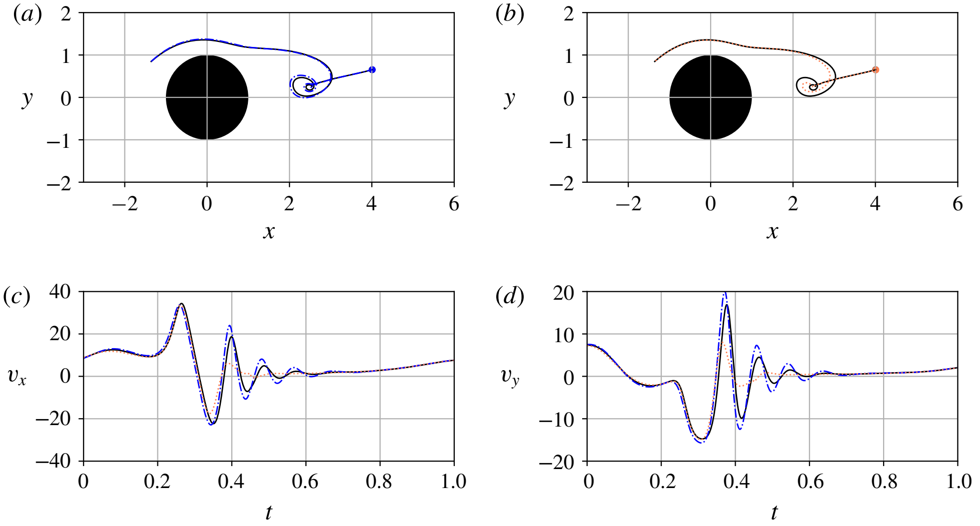

FIGURE 3. (*a*) 500-step trajectory predicted using the blended model for a von Kármán vortex flow trajectory compared with (*b*) the leading-order approximation. (*c*,*d*) Show predicted velocity versus time in the x and y directions. (Legend: truth ——; blended model — · —; leading-order slow manifold · · · · · ·).

target dimension for training and test flows. For the trained model to perform well, regions where the test flow has non-negligible probability density should also have high density in the training flow.

For single-step predictions, we generate reference trajectories from (2.2) under the same particle parameters ($\epsilon = 0.01$, $R = 1.55$) and time step (0.002). These trajectories are randomly initiated in a rectangular box upstream of the cylinder: $x_0 \sim \mathcal{U}[-2.1, -1.9]$, $y_0 \sim \mathcal{U}[-1, 1]$. The model is able to achieve a single-step MSE of 0.08, averaged over 50 test cases consisting of 550 time steps each. The performance is significantly better than the leading-order approximation alone, which records a MSE of 1.89 for the same data set. This result shows that the model we have obtained indeed generalizes well to new flow fields.

As for multistep predictions, we use a 50-step spin-up and predict for the 500 steps that immediately follow. Figure 3 shows an example prediction in terms of the resulting phase trajectory and velocity time series, compared with those given by the leading-order approximation. We observe that our blended model makes the biggest improvement in predicting how the particle travels with a shed vortex. It successfully captures the oscillation in the particle velocity as a result of the strong vorticity, which is not reflected in the leading-order approximations.

4.3. *Two-dimensional turbulent flow*

In the final application, we apply the blended model to an even more complex turbulent flow. The velocity field $\boldsymbol{u}$ for this flow is obtained as the numerical solution to the Navier–Stokes equation,

$$\partial_t \boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} = -\nabla p + \nu\Delta\boldsymbol{u} + \boldsymbol{f}, \quad \nabla\cdot\boldsymbol{u} = 0 \qquad (4.2a,b)$$

by a pseudo-spectral method with 128^2 Fourier modes. We use an explicit Runge–Kutta method for temporal integration. $\nu = 1/Re$ is the non-dimensional viscosity.

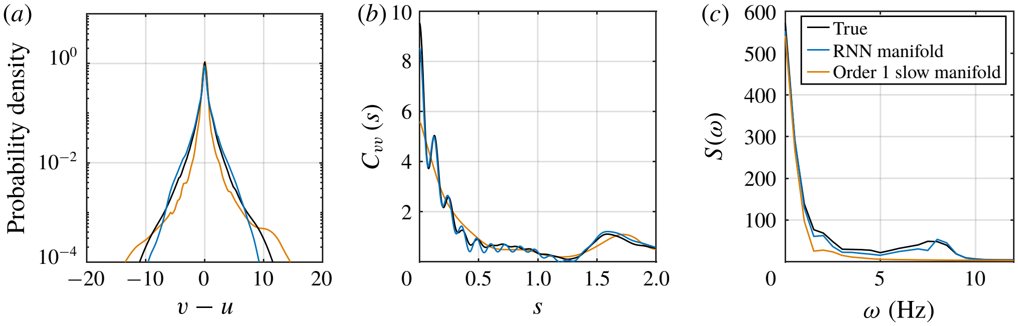

FIGURE 4. (*a*) Pdf for $v - u$, i.e. the velocity difference between the particle and flow, (*b*) autocorrelation function and (*c*) power spectral density for v as predicted by the true Maxey–Riley equation, multistep blended model and leading-order slow manifold approximations. Plotted quantities do not distinguish between x and y directions as the particle kinematics is roughly isotropic.

The time-independent forcing $\boldsymbol{f}$ has constant amplitude f_a and random phase, but only active for wavenumbers $\boldsymbol{k}$ on a circular ring, i.e. satisfying $k_{min}^2 \leqslant |\boldsymbol{k}|^2 < k_{max}^2$, where k_{min} and k_{max} are the inner and outer radii respectively. Equation (4.2) is solved on a spatial domain $(x, y) \in [0, 2\pi] \times [0, 2\pi]$ prescribed with periodic boundary conditions. We choose to validate the blended model for flow parameters $Re = 250$, $f_a = 1$ $k_{min} = 4$ and $k_{max} = 5$. Snapshots of the flow at the grid points are recorded every 2×10^3 time units. A linear interpolation scheme is employed to compute the flow velocities between the grid points.

For this flow we model trajectories of particles with $\epsilon = 0.05$ and $R = 1.55$. The particle inertia is increased from previous examples to raise higher-order effects. We train a model using data collected from the much simpler cellular flow with random coefficients: $A \sim \mathcal{U}[10, 20]$, $B \sim \mathcal{U}[0, 3]$ and $\omega \sim \mathcal{U}[0, 4\pi]$. These parameters are slightly different from the values used in the previous examples to ensure the global attractiveness of the underlying slow manifold. After training, the blended model achieves a single-step prediction MSE of 0.09, averaged over 1200 steps and 1000 test trajectories randomly initialized in the spatial domain. In contrast, MSE of the leading-order slow manifold approximation is 9.88.

For multistep prediction we use a 200-step spin-up stage. Computing the multistep prediction error for this turbulent flow is not meaningful because any small errors made would grow exponentially with time due to the presence of chaos. However, we may still use the model to calculate long-time statistics. Running with 5000 realizations for 1000 steps, the empirical probability density function (pdf) of the difference between particle and flow velocities (in both directions, as the flow turbulence is isotropic) is shown in figure 4(*a*). We observe that the blended model provides a much more improved estimation compared to the leading-order approximation in both the bulk and tail regions. In figure 4(*b*,*c*) we compare the autocorrelation function and power spectral density estimations, respectively, for particle velocities. The blended model is better, primarily due to its ability to capture correlation patterns in the presence of vortices, although it tends to predict longer times for particles to be drawn to vortex centres and results in slightly stronger oscillation in the autocorrelation function. Figure 5 shows snapshots of the true and model-predicted particle positions at $t = 0.8$ (end of spin-up period), 1.2 and

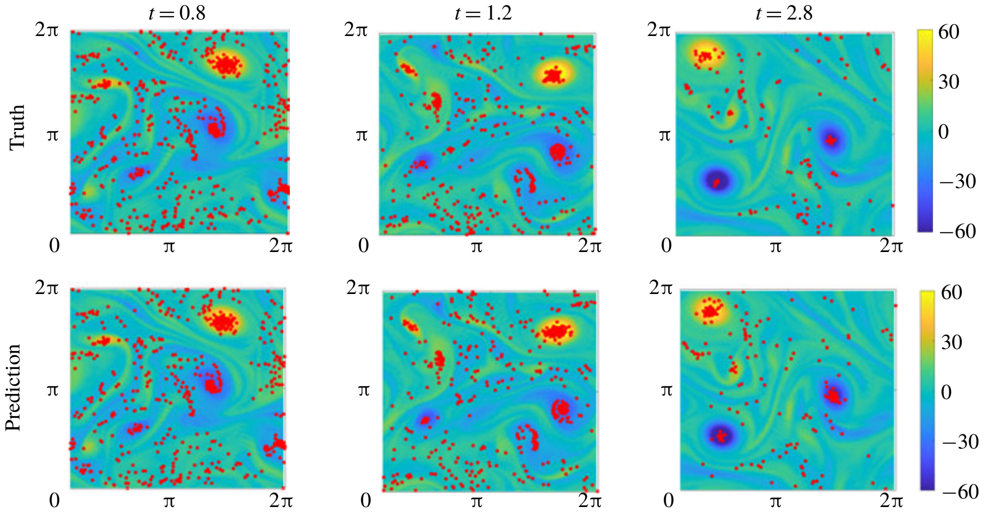

FIGURE 5. Particle position (marked by red dots) predicted by the Maxey–Riley equation (truth) and multistep blended model at time 0.8 (end of spin-up phase, zero error), 1.2 and 2.8, respectively. The flow vorticity field $\omega_z = \nabla \times \boldsymbol{u}$ is shown in the background. Particles are drawn towards vortices. Distributions of their positions are captured to good accuracy.

2.8, respectively. Similar characteristics are observed for both methods. In general, particles are drawn towards vortex centres because they are lighter than infinitesimal fluid elements ($R > 2/3$), as suggested by both the true MR and prediction models.

5. Conclusions

We have developed a blended, model-based, machine learning approach for describing the motion of small spherical particles in fluid flows. The proposed method takes advantage of the existence of an analytical model: a low-dimensional, globally attracting slow manifold of the Maxey–Riley equation. It also employs delay coordinates through a machine learning strategy to complement the analytical leading-order approximation of the manifold and improve the overall accuracy. We have demonstrated the applicability through three different flow scenarios. We showed that a kinematical model constructed from particle trajectories in a flow with relatively simple structures can successfully generalize to more complex – and even turbulent – flows with no additional readjustment. The resulting single-step velocity predictions are significantly improved from the leading-order estimates. More impressively, the multistep implementation of the model is able to provide decent estimates for velocity statistics, as well as the autocorrelation and power spectrum. Our future endeavours will focus on the learning of the bubble dynamics directly from DNS simulations. In this case we expect that more variables will be utilized in the machine learning model, including the bubble size and fluid field pressure.

Acknowledgements

We thank Mr. A. T. Charalampopoulos for providing the code for the generation of the turbulent velocity fields in the third example. This work has been supported through the ONR MURI grant N00014-17-1-2676.

References

BABIANO, A., CARTWRIGHT, J. H. E., PIRO, O. & PROVENZALE, A. 2000 Dynamics of a small neutrally buoyant sphere in a fluid and targeting in Hamiltonian systems. *Phys. Rev. Lett.* **84** (25), 5764–5767.

BENCZIK, I. J., TOROCZKAI, Z. & TÉL, T. 2002 Selective sensitivity of open chaotic flows on inertial tracer advection: catching particles with a stick. *Phys. Rev. Lett.* **89** (16), 164501.

CARTWRIGHT, J. H. E., FEUDEL, U., KÁROLYI, G., DE MOURA, A., PIRO, O. & TÉL, T. 2010 Dynamics of finite-size particles in chaotic fluid flows. In *Nonlinear Dynamics and Chaos: Advances and Perspectives*, pp. 51–87. Springer.

CROWE, C., TROUTT, T. & CHUNG, J. 1996 Numerical models for two-phase turbulent flows. *Annu. Rev. Fluid Mech.* **28**, 11.

DAITCHE, A. & TÉL, T. 2014 Memory effects in chaotic advection of inertial particles. *New J. Phys.* **16**, 073008.

HALLER, G. & SAPSIS, T. P. 2008 Where do inertial particles go in fluid flows? *Physica* D **237** (5), 573–583.

HOCHREITER, S. & SCHMIDHUEBER, J. 1997 Long short-term memory. *Neural Comput.* **9** (8), 1–32.

JUNG, C., TÉL, T. & ZIEMNIAK, E. 1993 Application of scattering chaos to particle transport in a hydrodynamical flow. *Chaos* **3**, 555–568.

MARCU, B., MEIBURG, E. & NEWTON, P. 1995 Dynamics of heavy particles in a Burgers vortex. *Phys. Fluids* **7**, 400.

MARTIN, J. & MEIBURG, E. 1994 The accumulation and dispersion of heavy particles in forced two-dimensional mixing layers: I. The fundamental and subharmonic cases. *Phys. Fluids* **6**, 1116.

MAXEY, M. R. & RILEY, J. J. 1983 Equation of motion for a small rigid sphere in a nonuniform flow. *Phys. Fluids* **26** (4), 883–889.

SAPSIS, T. P. & HALLER, G. 2008 Instabilities in the dynamics of neutrally buoyant particles. *Phys. Fluids* **20**, 17102.

SAPSIS, T. P. & HALLER, G. 2009 Inertial particle dynamics in a hurricane. *J. Atmos. Sci.* **66**, 2481–2492.

SAPSIS, T. P. & HALLER, G. 2010 Clustering criterion for inertial particles in two-dimensional time-periodic and three-dimensional steady flows. *Chaos* **20** (1), 017515.

SAPSIS, T. P., HALLER, G. & PENG, J. 2011 Predator Prey interactions in jellyfish feeding. *Bull. Math. Biol.* **73**, 1841–1856.

SAPSIS, T. P., OUELLETTE, N. T., GOLLUB, J. P. & HALLER, G. 2011 Neutrally buoyant particle dynamics in fluid flows: comparison of experiments with Lagrangian stochastic models. *Phys. Fluids* **23**, 093304.

SUDHARSAN, M., BRUNTON, S. L. & RILEY, J. J. 2016 Lagrangian coherent structures and inertial particle dynamics. *Phys. Rev.* E **93**, 033108.

TANG, L., WEN, F., YANG, Y., CROWE, C., CHUNG, J. & TROUTT, T. 1992 Self-organizing particle dispersion mechanism in a plane wake. *Phys. Fluids* A **4**, 2244.

TIO, K K., LINAN, A., LASHERAS, J. & GANAN-CALVO, A. 1993 On the dynamics of buoyant and heavy particles in a periodic Stuart vortex flow. *J. Fluid Mech.* **253**, 671.

VASILIEV, A. A. & NEISHTADT, A. I. 1994 Regular and chaotic transport of impurities in steady flows. *Chaos* **4**, 673.

VLACHAS, P. R., BYEON, W., WAN, Z. Y., SAPSIS, T. P. & KOUMOUTSAKOS, P. 2018 Data-driven forecasting of high dimensional chaotic systems with LSTM networks. *Proc. R. Soc.* A **474**, 20170844.

WAN, Z. Y., VLACHAS, P. R., KOUMOUTSAKOS, P. & SAPSIS, T. P. 2018 Data-assisted reduced-order modeling of extreme events in complex dynamical systems. *PLoS One* **13** (5), e0197704.