

Journal Pre-proof

Tracking in-silico Lagrangian sensors in a lab-scale stirred tank reactor

Vamika Rathi, Fatima Sehar, Finn Sommer, Sebastian Götschel,
Eike Steuwe, Alexandra von Kameke, Daniel Ruprecht



PII: S0098-1354(26)00363-7
DOI: <https://doi.org/10.1016/j.compchemeng.2026.109911>
Reference: CACE 109911

To appear in: *Computers and Chemical Engineering*

Received date : 11 June 2026

Revised date : 7 September 2026

Accepted date : 17 September 2026

Please cite this article as: V. Rathi, F. Sehar, F. Sommer et al., Tracking in-silico Lagrangian sensors in a lab-scale stirred tank reactor. *Computers and Chemical Engineering* (2026), doi: <https://doi.org/10.1016/j.compchemeng.2026.109911>.

This is a PDF of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability. This version will undergo additional copyediting, typesetting and review before it is published in its final form. As such, this version is no longer the Accepted Manuscript, but it is not yet the definitive Version of Record; we are providing this early version to give early visibility of the article. Please note that Elsevier's sharing policy for the Published Journal Article applies to this version, see: <https://www.elsevier.com/about/policies-and-standards/sharing#4-published-journal-article>. Please also note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2026 Published by Elsevier Ltd.

Highlights

Tracking in-silico Lagrangian sensors in a lab-scale stirred tank reactor

Vamika Rathi, Fatima Sehar, Finn Sommer, Sebastian Götschel, Eike Steuwe, Alexandra von Kameke, Daniel Ruprecht

- We propose a framework to track Lagrangian sensors based on accelerometer and magnetometer readings
- An extended Kalman filter and a particle filter are proposed to fuse sensor readings with a bespoke dynamic model
- Experiments for in-silico replicas of a recent Lagrangian sensor design show that trajectories can be reconstructed with [mean](#) relative errors below [3%](#)

Tracking in-silico Lagrangian sensors in a lab-scale stirred tank reactor

Vamika Rathi^{a,*}, Fatima Sehar^a, Finn Sommer^a, Sebastian Götschel^a, Eike Steuwe^b, Alexandra von Kameke^b, Daniel Ruprecht^a

^a*Chair Computational Mathematics, Institute of Mathematics, Hamburg University of Technology, Hamburg, Germany*

^b*Heinrich-Blasius Institute, Faculty of Engineering and Computer Science, Hamburg University of Applied Sciences, Hamburg, Germany*

Abstract

Lagrangian sensors have shown promise to improve operator awareness of conditions inside a chemical reactor but three-dimensional tracking remains a mostly unsolved challenge. We explore a setup where in-silico sensors, based on a recently proposed real-world design, are tracked using data from an accelerometer and magnetometer available from a built-in inertial measurement unit. Filtering algorithms, using a bespoke dynamical model, are used to process these readings into position estimates. We compare tracking performance of an extended Kalman filter, a particle filter and the unscented Kalman filter implemented in the `pykalman` library. Our numerical experiments track in-silico particles moving in an analytically given three-dimensional vortex as well as in the experimentally measured flow-field of a lab-scale stirred tank reactor. Using the Maxey-Riley-Gatignol equations for the movement of inertial particles as ground-truth, we demonstrate that [the extended Kalman filter and the particle filter reconstruct](#) trajectories from noisy synthetic data with [mean](#) relative errors below [3%](#).

Keywords: Lagrangian sensors, stirred-tank reactor, Maxey-Riley-Gatignol equations, Kalman filter, Particle filter

*This project is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – SFB 1615 – 503850735.

*Corresponding author

1. Introduction

Lagrangian sensors (LS) are sensors that travel freely in a liquid and provide measurements along their trajectory. By contrast, Eulerian sensors are fixed and provide measurements over time at a single point in space. LS are used in a variety of applications. Examples are drifters in oceanography [1], coastal engineering [2] or to design pipes that avoid injury to fishes [3]. A widely known example of Lagrangian sensors are ingestible capsules used in medicine, that travel through the gastrointestinal tract – a highly acidic chemical reactor – and transmit measurements along the way. Designs range from the video capsule endoscope [4] to capsules that sense gases produced by the gut microbiota [5]. In chemical process engineering, Lagrangian sensors have been explored for decades as means to provide data about the interior state of chemical reactors, starting with the concept of a “radio pill” proposed in 1969 [6, 7]. Eulerian sensors, installed at the walls of a vessel, only give an incomplete picture of the reactions happening inside, in particular in bio-reactors where homogeneous mixing is difficult to achieve because too vigorous stirring would damage the cells [8]. Various devices have been proposed and studied over the years, see Bisgaard et al. (2020) for a comprehensive survey and their Figure 2 for a timeline of developments [9]. LS have also been conceptualized as an important part of a chemical reactor’s digital twin [10]. A broader perspective on flow-following sensor technology has emphasized its potential for validating CFD simulations and characterizing industrial flows [11].

LS have been equipped with a range of sensors to measure temperature, pressure, acceleration, pH, dissolved oxygen or magnetic fields [9]. There are demonstrations in practical settings that show that LS can provide a range of useful information about the interior state of a chemical reactor, for example average residence times in vertical compartments, but collected data remain statistical rather than spatially or temporally resolved [8, 12, 13, 14, 15, 16]. A key obstacle is that localization remains a challenge. In experimental setups, it is possible to track particles by some distinct property like color in video tracking, an emitted radio-signal, electrical resistance tomography, magnetic permeability or radioactivity in positron emission particle tracking (PEPT) [17, 18, 19, 20]. While extremely useful in a lab setting, these techniques cannot be used in standard operation. Pressure readings and the assumption of hydro-static pressure can provide approximate vertical positioning and allow for the determination of averaged quantities with some

accuracy [12, 17, 18] but do not provide information about horizontal position.

Fully time and space resolved tracking is complicated by the fact that commonly sized LS do not truly follow the flow but experience inertial effects depending on their size and velocity gradients in the fluid [21]. How closely a LS follows the flow is determined by its Stokes number, a non-dimensional parameter that represents the ratio of the particle response time to the characteristic time scale of the fluid [21]. Bisgaard et al. (2022) show that LS can range in Stokes numbers from $\mathcal{O}(10^{-2})$ up to $\mathcal{O}(10)$, depending on their size and the viscosity of the carrier liquid. There is evidence that particles with Stokes number $\mathcal{O}(10^{-1})$ and smaller can reasonably be considered flow-following with errors of 1% or less [22, §5.3.1]. Because of the required instrumentation, however, particular LS with broader capabilities are larger and fall outside of this range.

Some sensor readings in an LS can be used to track its position, in particular readings from an IMU, magnetometer or pressure sensor. Acceleration data from an inertial measurement unit IMU can be used for dead reckoning, but, on its own, suffers from rapidly accumulating drift errors over time. Magnetic fields, either from the Earth or an artificial source, can act as beacons, and their measurements can be fused with IMU data for more robust tracking. A magnetic field has been used, for example, to detect when a particle is at the apex of a chemical reactor to derive circulation times [18]. Buntkiel et al. (2023) study the performance of triaxial accelerometer in combination with a magnetometer measuring Earth's magnetic field fused via a discrete-time Kalman filter for positioning [23]. Buntkiel et al. validate reconstructed mean radial accelerations against measurements and, in their conclusions, propose to use an artificial time-dependent magnetic field to improve performance. The idea of using an artificial magnetic field to track a LS called Sens-O-Sphere was proposed by Lange [24] but seems to have not yet been realized.

Motion tracking using readings from magnetometers and inertial sensors has been proposed to reduce computational requirements while maintaining orientation accuracy through EKF-based correction steps [25]. An IMU/magnetometer-based indoor positioning system demonstrated that inertial data can be fused with magnetic field information to reduce drift in portable tracking systems [26]. Broader reviews of motion capture technologies in industrial environments highlight a growing interest in tracking freely moving bodies, but note that most systems rely on external references and

kinematic motion models rather than physics-based dynamics [27]. Wu et al. showed that IMU data fused with readings from an artificial dipole magnetic field via an extended Kalman filter can provide robust indoor positioning of a mobile cart [28].

In this paper, we follow up on the suggestion by Buntkiel et al. [23] to use an artificial magnetic field and demonstrate that such a setup can track in-silico LS in the experimentally determined flow field of a lab-scale stirred tank reactor using a simulated artificial magnetic field. In contrast to previous approaches, we use a physics-based state transition model in an extended Kalman filter (EKF) to account for the added mass and drag that LSs experience. For validation, we employ the same dynamical model in a particle filter and compare its performance against that of the EKF. To illustrate the benefit from using a physics-based transition model, we also compare tracking performance against the unscented Kalman filter (UKF) from the `pykalman` library [29]. All filters use the same dynamics and measurement models. All the tracking results shown in the paper can be reproduced using the published code [30].

Kalman type filters are one branch of a broader family of model based estimators. The classical alternative is output error injection, in which a copy of the process model is corrected by the output residual through a fixed gain [31], with nonlinear extensions and established use in chemical process engineering [32, 33]. A second alternative is moving horizon estimation, which solves a constrained least squares problem over a receding window and can enforce physical bounds on the state [34]. We nevertheless adopt recursive Bayesian filters for two reasons specific to Lagrangian sensor tracking. First, position is observed only through the artificial dipole field, whose sensitivity varies strongly across the tank, so a gain well conditioned in one region is poorly suited to another, whereas the innovation covariance rescales the correction automatically. Second, the dominant error source is the unmodelled physics of the MaRGE truncation rather than sensor noise. This error is represented explicitly as process noise and propagated into a reported uncertainty on the reconstructed trajectory by a stochastic formulation, for which an observer offers no equivalent. Experiments with an extended Luenberger observer [35] with constant, pre-defined gain were unsuccessful in tracking the particles. While these results are not shown in the paper, they can be reproduced with the accompanying code [30].

The remainder of the paper is organised as follows. §2 describes the tracking framework, the Maxey-Riley-Gatignol equations used to generate

ground-truth trajectories, the two velocity fields, and the three filters. §3 presents tracking results for in-silico sensors in an analytically given three-dimensional vortex and in the experimentally measured flow field of a lab-scale stirred tank reactor. §4 draws conclusions and discusses limitations and directions for future work.

2. Methodology

In §2.1 we describe the proposed tracking setup, including the physical characteristics of our in-silico particles. §2.2 introduces the equations governing inertial particle motion that we use to create ground-truth trajectories. In reduced form, these equations also serve as dynamic models used for tracking. §2.3 describes the two velocity field required as input to the dynamical equations described in §2.2, a 3D vortex and an experimental flow field of a lab-scale stirred tank reactor. The experimental setup that was used to obtain the second velocity field is also briefly described. §2.4 introduces the used extended Kalman filter while §2.5 introduces the particle filter, including the necessary modifications to ensure accurate tracking like adaptive tempering and innovations. Finally, §2.6 describes the unscented Kalman filter, which serves as a further point of comparison and uses the same dynamic and measurement models as the other two filters.

2.1. Tracking framework

Figure 1 shows a sketch of the proposed setup. A magnetic field source is placed above the reactor and generates a magnetic dipole field

$$B(\mathbf{r}) = \frac{\mu_0}{4\pi r^5} (3(\mathbf{m} \cdot \mathbf{r})\mathbf{r} - r^2 \mathbf{m}) \quad (1)$$

which is commonly used for magnetic field base positioning systems [36]. Here, μ_0 is the magnetic permeability of the free space, $\mathbf{r}$ is the position vector originating from the magnetic field source, $r = \|\mathbf{r}\|$ the distance and $\mathbf{m}$ the magnetic moment of the coil generating the field, which is orthogonal to the coil surface. Our in-silico particles are based on the real-world design proposed by Geläsčius et al. (2026) [37]. The particles contain an IMU with an accelerometer and vector magnetometer that can measure the strength and direction of the magnetic field [38]. Data transmission will be realized using low-power communication methods suited to a reactor environment [37]

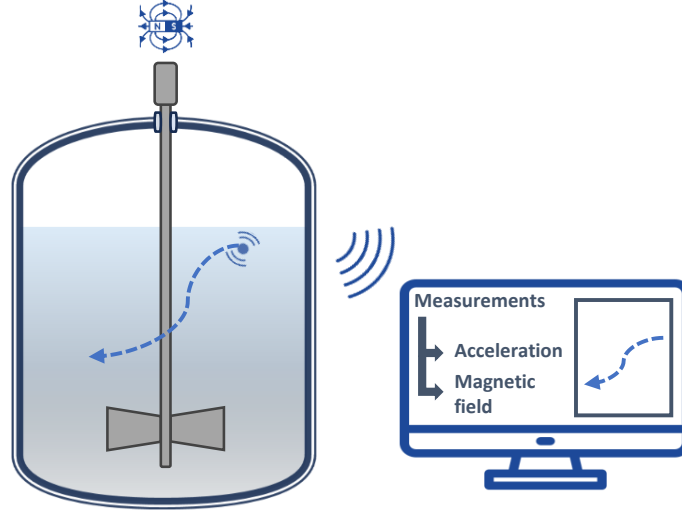


Figure 1: Illustration of a sensor inside a stirred tank reactor and position of the magnetic field source.

but, for the purpose of this paper, is assumed to work reliably except for artificially added Gaussian noise.

Note that many LS also feature pressure sensors that can be used for approximate vertical positioning by assuming hydrostatic pressure in the reactor vessel [12]. Introducing [such sensors](#) as an additional source of information into the filtering algorithms would be relatively straightforward, but requires some modifications and is thus left for future work. While we test our approach for experimental flow-field data, our particles are in-silico, that is, simulated replicas, and the ground-truth trajectories we reconstruct are simulated. However, the physical parameters [of the in-silico particles](#) summarized in Table 1 match those of the proposed design [37]. The corresponding fluid parameters of the experimental water-based setup are also included in Table 1.

2.2. Maxey-Riley-Gatignol Equations

Because of their size, [LS experience significant inertial effects and](#) cannot be considered passive tracers [8]. Motion of small rigid spherical particles in a fluid can be described by the Maxey-Riley-Gatignol equation (MaRGE) [39, 40]. [MaRGE is not exact but an asymptotic model, valid for a particle that is small compared to the length scales over which the carrier flow varies](#)

Parameter	Symbol	Value
Particle radius	a	0.0025 m
Particle density	ρ_p	1010 kg/m ³
Fluid density	ρ_f	998 kg/m ³
Fluid kinematic viscosity	ν	$1.004 \times 10^{-6} \text{ m}^2 \text{ s}^{-1}$
Fluid characteristic length	L	0.130 m
Fluid characteristic velocity	U	0.07 m s^{-1}

Table 1: Table of physical parameters of in-silico particles and the experimental flow field.

and for a small particle Reynolds number. We use the form without Faxén corrections and neglect the feedback of the particle on the carrier flow. Let $\mathbf{v} = \dot{\mathbf{x}}$ be the absolute velocity of a particle at position $\mathbf{x}$ with radius a and density ρ_p . The resulting form of MaRGE, which is the one commonly used in the literature, reads in dimensionless form

$$\frac{d\mathbf{v}}{dt} = R \frac{D\mathbf{u}}{Dt} - \frac{R}{S}(\mathbf{v} - \mathbf{u}) - R\sqrt{\frac{3}{S\pi}} \frac{d}{dt} \int_{t_0}^t \frac{1}{\sqrt{t-\tau}}(\mathbf{v} - \mathbf{u}) d\tau - (1-R)\mathbf{G}, \quad (2)$$

using a characteristic time T and characteristic velocity U to non-dimensionalize, following the approach by Daitche [41]. Here, t_0 is the initial time, $\mathbf{u}(\mathbf{x}, t)$ is the fluid velocity at the particle position and R , S and $\mathbf{G}$ are the dimensionless parameters

$$R = \frac{3\rho_f}{\rho_f + 2\rho_p}, \quad S = \frac{1}{3} \frac{a^2/\nu}{T}, \quad \mathbf{G} = \frac{T}{U} \mathbf{g}, \quad (3)$$

where ρ_f is the density of the fluid, ν is the kinematic viscosity of the fluid and $\mathbf{g}$ is the gravitational acceleration vector. The material derivative

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \quad (4)$$

in (2) is calculated analytically for the vortex field. For the experimental field, where only discrete data points are available, the gradient is computed by spline interpolation and the temporal derivative by linear interpolation.

Using the parameters in Table 1 and a characteristic time scale of $T = L/U \approx 1.86 \text{ s}$, we get $R = 0.992$ and $S = 1.12$. Throughout the paper, we refer to the parameter S as the Stokes number as it compares the particle relaxation time a^2/ν to the flow timescale T [42] but note that slightly different uses exist in the literature. We solve (2) numerically using a linear

multi-step method developed by Daitche [41] and generalized to 3D by Rathi et al. [43]. For various simplified flow fields, analytical solutions for S ranging from 0.01 to 1 are provided by Prasath et al. (2019) [44].

The purpose of this paper is to demonstrate that the proposed framework in principle allows tracking a particle moving inside the fluid. We use (2) including the Basset force modelled by the integral to generate ground-truth trajectories and synthetic data, to which artificial 5% Gaussian noise is added. The two filtering algorithms introduced below use a continuous process model that ignores the Basset force to reconstruct the ground-truth trajectories from the noisy synthetic data. In essence, our setup introduces a controlled model error for in-silico particles. By contrast, validation for real-world particles, which is planned for future work, would face additional challenges as the model error when using MaRGE to predict inertial particle motion is not yet fully understood which would make the physics-based dynamic models in the filters less accurate.

2.3. Velocity fields

The key assumption in MaRGE is that the impact of the particle's movement on the carrier liquid is confined to the integral term in (2), which models wake effects and accounts for temporal delays in boundary layer development [45]. Therefore, the fluid flow field $\mathbf{u}$ in (2) needs to be prescribed and is not modelled. We consider two different flow fields, an analytically given 3D vortex and an experimentally determined velocity field in a lab-scale stirred-tank reactor.

Vortex flow. Our 3D vortex is an extension of a simple two-dimensional vortex proposed by Candelier [46],

$$\mathbf{u}(x, y, z, t) = \omega(z, t) \begin{bmatrix} -y \\ x \\ 0 \end{bmatrix}, \quad (5)$$

with an angular frequency $\omega(z, t) = \omega_0 + \alpha \sin^2(z) \cos^2(t)$ that depends on z and t . We use $\omega_0 = 4$ and $\alpha = 0.2$. The z component for this field is zero, which means there is no velocity in the vertical direction but the particle will still sink or rise due to gravity.

Lab-scale stirred tank reactor. The experimental flow field is obtained from a lab scale reactor with diameter 130 mm and height 230 mm. The data for the experimental flow field were provided by the group of Alexandra von Kameke, using a 3 litre stirred tank reactor with two Rushton turbines [47]. A *shake-the-box* [48] method was used to compute a Lagrangian representation of the flow field from a large number of images of passive tracer particles taken by high-speed cameras. Velocities were mapped to an Eulerian grid using the scattered interpolant method implemented in Matlab [49]. Data covers a period of 2 seconds with a temporal resolution of 0.002 seconds for a total of 1001 snapshots. For efficiency, we consider only every fifth snapshot and discard the rest. Linear interpolation is used to produce velocity data between snapshots. To obtain a continuous representation $\hat{\mathbf{u}}$ of the measured fluid field that can be evaluated for any particle position (x, y, z) , the data on the Eulerian grid is interpolated using cubic spline interpolation provided by the *RegularGridInterpolator* of the SciPy interpolation sub-package.¹ The spatial gradient in (4) is computed by using finite differences provided by the Python package *findiff* [50].

To obtain the interpolated fluid velocity at some arbitrary time t , we use the interpolation objects $\hat{\mathbf{u}}(x, y, z, t_i)$ and $\hat{\mathbf{u}}(x, y, z, t_{i+1})$ provided by SciPy for two nearest times $t_i \leq t \leq t_{i+1}$ for which data is available. The velocity $\hat{\mathbf{u}}(x, y, z, t)$ is then calculated by linear interpolation.

$$\hat{\mathbf{u}}(x, y, z, t) = \hat{\mathbf{u}}(x, y, z, t_i) + \frac{\hat{\mathbf{u}}(x, y, z, t_{i+1}) - \hat{\mathbf{u}}(x, y, z, t_i)}{t_{i+1} - t_i}(t - t_i).$$

The temporal partial derivative in (4) is approximated by

$$\frac{\partial \mathbf{u}}{\partial t} \approx \frac{\hat{\mathbf{u}}(x, y, z, t_{i+1}) - \hat{\mathbf{u}}(x, y, z, t_i)}{t_{i+1} - t_i}.$$

Trajectories. Figure 2 shows two trajectories for our in-silico 5 mm diameter device with the physical parameters given in Table 1. The device is denser than the fluid, which causes it to sink into the vortex, as shown in the left panel of Figure 2. In line with previous results, neglecting the Basset force modelled by the history term causes the particle to sink much faster [42, 46]. For the experimental flow field, the impact of ignoring the Basset term is

¹<https://docs.scipy.org/doc/scipy/reference/interpolate.html>

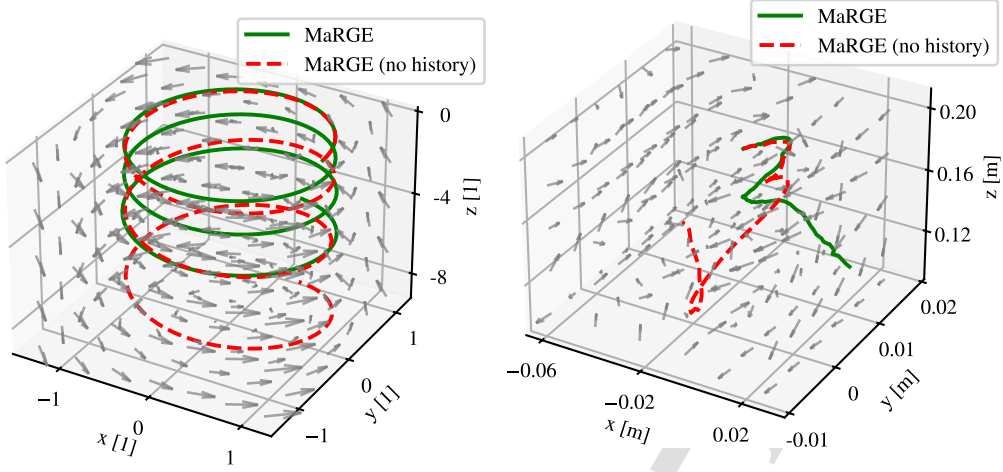


Figure 2: The effect of history term on the trajectory of inertial particle. The figure on the left is for the particle in analytical flow field for time $t \in [0, 5]$. The right figure is for the particle in experimental flow field from a stirred tank reactor for time $t \in [0, 2]$ sec. Trajectories obtained by MaRGE with (green, solid) and without (red, dashed) history term are plotted.

even more drastic. Due to the lack of inertia, the particle moves at a completely different path. This difference shows that ignoring the Basset force in the dynamic models of the filters introduces a controlled but significant model error which makes the reconstruction of the ground-truth from noisy synthetic data very challenging.

2.4. Extended Kalman filter

This section describes how to adopt an Extended Kalman filter (EKF)[51, 52] for the described tracking setup. EKF is the most commonly used non-linear version of the Kalman Filter. The nonlinear system equations are approximated through linearization about the current state estimate. As process model we use $d\mathbf{x}(t)/dt = \mathbf{v}(t)$ and (2) without the history term,

$$\frac{d\mathbf{v}}{dt} = R \frac{D\mathbf{u}}{Dt} - \frac{R}{S}(\mathbf{v} - \mathbf{u}) - (1 - R)\mathbf{G}. \quad (6)$$

Particle acceleration is denoted as $\mathbf{a} = \frac{d\mathbf{v}}{dt}$. We write the system state at some time point t_k as $\mathbf{S}_k = [\mathbf{x}_k \ \mathbf{v}_k \ \mathbf{a}_k]^T$, where $\mathbf{x}$, $\mathbf{v}$ and $\mathbf{a}$ are the 3D position, velocity and acceleration vectors of the LS. Since explicit Euler is

unconditionally unstable for second-order problems like (6), we discretize the process model using symplectic Euler

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \Delta t \mathbf{a}_k, \quad (7a)$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta t \mathbf{v}_{k+1} = \mathbf{x}_k + \Delta t \mathbf{v}_k + \Delta t^2 \mathbf{a}_k, \quad (7b)$$

where Δt is the time step size and acceleration is computed from

$$\mathbf{a}_k = R \left. \frac{D\mathbf{u}}{Dt} \right|_{t_k} - \frac{R}{S}(\mathbf{v}_k - \mathbf{u}_k) - (1 - R)\mathbf{G}. \quad (8)$$

This implementation allows to use a numerical time-step Δt that is smaller than the sensor frequency, which allows for better control of the numerical error. We use the following notations in the description of the EKF:

$\mathbf{S}_{k+1,k}$ - predicted estimate of the system state at time t_{k+1} calculated at time t_k .

$\mathbf{P}_{k+1,k}$ - predicted covariance of the system state at time t_{k+1} calculated at time t_k .

$\mathbf{S}_{k+1,k+1}$ - updated estimate of the system state at time t_{k+1} .

$\mathbf{P}_{k+1,k+1}$ - updated covariance of the system state at time t_{k+1} .

We can now write the state extrapolation equation as

$$\mathbf{S}_{k+1,k} = \begin{bmatrix} \mathbf{x}_{k+1,k} \\ \mathbf{v}_{k+1,k} \\ \mathbf{a}_{k+1,k} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{k,k} + \Delta t \mathbf{v}_{k,k} + \Delta t^2 \mathbf{a}_{k,k} \\ \mathbf{v}_{k,k} + \Delta t \mathbf{a}_{k,k} \\ \mathbf{a}_{k,k} \end{bmatrix} =: \mathbf{f}(\mathbf{S}_{k,k}). \quad (9)$$

While this model appears linear in the state variables, the nonlinearity enters through the acceleration term $\mathbf{a}_{k,k}$, which is obtained from (8). Specifically, the fluid velocity $\mathbf{u}(\mathbf{x}, t)$ and its derivatives are nonlinear functions of the particle position $\mathbf{x}$ making the state transition function $\mathbf{f}$ nonlinear. The covariance extrapolation reads

$$\mathbf{P}_{k+1,k} = \mathbf{F}_k \mathbf{P}_{k,k} \mathbf{F}_k^T + \mathbf{Q}, \quad (10)$$

where $\mathbf{F}_k = \frac{\partial \mathbf{f}}{\partial \mathbf{S}}$ is the Jacobian of the state transition matrix and $\mathbf{Q}$ is the process noise covariance matrix

$$\mathbf{Q} = \begin{bmatrix} \Delta t^4 \mathbf{I}_3 & \Delta t^3 \mathbf{I}_3 & \Delta t^2 \mathbf{I}_3 \\ \Delta t^3 \mathbf{I}_3 & \Delta t^2 \mathbf{I}_3 & \Delta t \mathbf{I}_3 \\ \Delta t^2 \mathbf{I}_3 & \Delta t \mathbf{I}_3 & \mathbf{I}_3 \end{bmatrix} \sigma_a^2, \quad (11)$$

where $\mathbf{I}_3$ is the 3×3 identity matrix and σ_a^2 is the acceleration variance of the dynamic model. The process noise covariance matrix is obtained for the dynamic model using the procedure by Becker [52]. The extrapolation equations predict the state variables and the covariance. Since we measure acceleration and magnetic field, the observation operator is

$$\mathbf{h}(\mathbf{S}) = \begin{bmatrix} \mathbf{a} \\ \mathbf{B}(\mathbf{x}) \end{bmatrix}, \quad (12)$$

where $\mathbf{B}$ is the magnetic field from (1). Thus, the measurement equation is

$$\mathbf{Z}_k = \mathbf{h}(\mathbf{S}_k). \quad (13)$$

The measurement covariance matrix $\mathbf{R}_k$ is assumed to be constant

$$\mathbf{R}_k = \begin{bmatrix} \sigma_{a_m}^2 \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \sigma_{B_m}^2 \mathbf{I}_3 \end{bmatrix}, \quad (14)$$

where $\sigma_{a_m}^2$ is the acceleration measurement variance, $\sigma_{B_m}^2$ is the magnetic measurement variance and $\mathbf{0}_3$ the 3×3 zero matrix. The Kalman gain is calculated as

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1,k} \frac{\partial \mathbf{h}^T}{\partial \mathbf{S}} \left(\frac{\partial \mathbf{h}}{\partial \mathbf{S}} \mathbf{P}_{k+1,k} \frac{\partial \mathbf{h}^T}{\partial \mathbf{S}} + \mathbf{R}_k \right)^{-1}, \quad (15)$$

where $\frac{\partial \mathbf{h}}{\partial \mathbf{S}}$ is the Jacobian of the nonlinear observation operator $\mathbf{h}(\mathbf{S})$ with respect to the system state. Since the first three rows of $\mathbf{h}(\mathbf{S})$ are just acceleration components and the magnetic field only depends on positions, we have

$$\frac{\partial \mathbf{h}}{\partial \mathbf{S}} = \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \\ \frac{\partial \mathbf{B}}{\partial \mathbf{x}} & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix}. \quad (16)$$

The Jacobian $\frac{\partial \mathbf{B}}{\partial \mathbf{x}}$ of the dipole magnetic field model is computed via automatic differentiation (AD) using the forward mode AD implementation in JAX [53]. Finally, the state update and covariance update equations, given the $(k+1)$ th measurement $\tilde{\mathbf{Z}}_{k+1}$, are

$$\mathbf{S}_{k+1,k+1} = \mathbf{S}_{k+1,k} + \mathbf{K}_{k+1}(\tilde{\mathbf{Z}}_{k+1} - \mathbf{h}(\mathbf{S}_{k+1,k})), \quad (17)$$

$$\mathbf{P}_{k+1,k+1} = \left(\mathbf{I}_9 - \mathbf{K}_{k+1} \frac{\partial \mathbf{h}}{\partial \mathbf{S}} \right) \mathbf{P}_{k+1,k} \left(\mathbf{I}_9 - \mathbf{K}_{k+1} \frac{\partial \mathbf{h}}{\partial \mathbf{S}} \right)^T + \mathbf{K}_{k+1} \mathbf{R}_{k+1} \mathbf{K}_{k+1}^T. \quad (18)$$

2.5. Particle filter

For comparison and cross-validation, we compare the EKF against a particle filter. Particle filters (PF) are sequential Monte Carlo methods for state estimation that approximate the filtering distribution by associating weights with random samples (called particles, not to be confused with our in-silico LS) [54, 55]. We implement a bootstrap PF which samples from the process model and uses the measurement likelihood as weights [56]. The posterior distribution at time step k , modelling the probability of the particle being in a certain state, is represented by a set of N_p weighted particles. In contrast to the EKF, the PF does not include acceleration in the state explicitly so that the state vector in the PF is defined as $\mathbf{X}_k = [\mathbf{x}_k \ \mathbf{v}_k]^T$. Let $\mathbf{X}_k^{(i)}$ denote the i -th particle's state with weight (probability) $w_k^{(i)}$. The final state estimate delivered by the PF is the weighted mean

$$\hat{\mathbf{X}}_k = \sum_{i=1}^{N_p} w_k^{(i)} \mathbf{X}_k^{(i)} \quad (19)$$

of the particle ensemble. Weights and particle states are continuously updated as new measurements become available.

Particles for the filter are initialized by sampling from a Gaussian distribution centred at the initial estimate $(\mathbf{x}_0, \mathbf{v}_0)$ with standard deviation $\sigma_{\text{init}} = 0.05$. The state propagation equations used to update the state of the particle with index i are the same discretization of MaRGE

$$\mathbf{v}_{k+1}^{(i)} = \mathbf{v}_k^{(i)} + \Delta t \mathbf{a}_k^{(i)} + \mathbf{n}_{v,k}^{(i)}, \quad (20)$$

$$\mathbf{x}_{k+1}^{(i)} = \mathbf{x}_k^{(i)} + \Delta t \mathbf{v}_{k+1}^{(i)} + \mathbf{n}_{x,k}^{(i)}, \quad (21)$$

that is used in the EKF (7), except for the addition of process noise $\mathbf{n}_{v,k}^{(i)}, \mathbf{n}_{x,k}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ sampled independently for each particle i at each time step k . As in the EKF, acceleration is computed from the particle state by (8) and the numerical timestep Δt can be chosen to be smaller than the sensor frequency to reduce the discretization error. The process noise covariance matrix

$$\mathbf{Q} = \begin{bmatrix} \Delta t^4 \mathbf{I}_3 & \Delta t^3 \mathbf{I}_3 \\ \Delta t^3 \mathbf{I}_3 & \Delta t^2 \mathbf{I}_3 \end{bmatrix} \sigma_a^2. \quad (22)$$

is based on the discrete white-noise acceleration model by Becker [52].

The PF maps the particle state to measurements via

$$\mathbf{a}_{\text{pred}}^{(i)} = \mathbf{a}_k^{(i)}, \quad (23)$$

$$\mathbf{B}_{\text{pred}}^{(i)} = \mathbf{B}(\mathbf{x}_k^{(i)}). \quad (24)$$

where $\mathbf{a}_k$ is the acceleration as in (8) and $\mathbf{B}(\mathbf{x}_k^{(i)})$ is the magnetic field from (1) at the particle position $\mathbf{x}_k^{(i)}$.

Because acceleration and magnetic field data differ in physical units and magnitude, we normalize both measurement residuals (also called innovations) by their respective uncertainties to ensure that they make comparable contribution to a particle's weight [52]. Acceleration innovations are normalized by combining the empirical particle spread of the measurement noise

$$\mathcal{Z}_{\text{acc},j}^{(i)} = \frac{a_{\text{pred},j}^{(i)} - \mathcal{Z}_{\text{acc},j}}{\sqrt{\sigma_{\text{ensemble},j}^2 + \sigma_{\text{meas}}^2}}, \quad (25)$$

where $\sigma_{\text{ensemble},j}$ is the standard deviation of particle predictions in component $j \in \{x, y, z\}$. This adaptive scaling down-weights acceleration measurements when particle predictions are inconsistent (large σ_{ensemble}), reflecting increased model uncertainty.

To ensure that magnetic field measurements remain valuable even when particles are spatially dispersed, magnetic field innovations are only scaled relative to field magnitude

$$\mathcal{Z}_{\text{mag},j}^{(i)} = \frac{\mathbf{B}_{\text{pred},j}^{(i)} - \mathcal{Z}_{\text{mag},j}}{\sigma_B \cdot |\mathcal{Z}_{\text{mag},j}|}, \quad (26)$$

where σ_B represents the relative measurement uncertainty of the magnetometer.

We assume conditional independence between sensor modalities and combine their contributions through multiplicative likelihood functions [57, 58]. Particle weights are updated via importance sampling

$$w_k^{(i)} \propto w_{k-1}^{(i)} p(\mathcal{Z}_{\text{acc}} | X_k^{(i)}) p(\mathcal{Z}_{\text{mag}} | X_k^{(i)}) \quad (27)$$

which becomes additive after applying a logarithm. Note that $\mathcal{Z}_{\text{acc}}, \mathcal{Z}_{\text{mag}} \in \mathbb{R}^3$ are vectors with components given by (25) and (26).

Resampling. Particle filters can suffer from ensemble degeneracy, when all but a very small number of particles have negligible importance weights. This represents essentially a collapse of particle diversity and leads to a substantial amount of superfluous computation, where particles keep getting updated that no longer contribute meaningfully to the overall estimate (19). Particle weight degeneracy can be measured using the effective sample size

$$\text{ESS} = \frac{1}{\sum_{i=1}^{N_p} (w_k^{(i)})^2}. \quad (28)$$

We use adaptive tempering of the likelihood function to help maintain particle diversity [59]

$$w_k^{(i)} \propto w_{k-1}^{(i)} \exp\left(\frac{\ell^{(i)}}{\tau}\right), \quad (29)$$

where $\tau \geq 1$ is the adaptive temperature parameter and is doubled iteratively until $\text{ESS} \geq 0.5N_p$ (number of particles) or $\tau = 64$. Here, $\ell^{(i)}$ is a fusion parameter introduced as

$$\ell^{(i)} = (1 - \gamma)\ell_{\text{acc}}^{(i)} + \gamma\ell_{\text{mag}}^{(i)}, \quad (30)$$

with $\gamma \in [0, 1]$ to explore different weighting between accelerometer and magnetometer readings. However, in experiments not documented here we found that equal weighting by using $\gamma = 0.5$ works well and no tuning is required. The log-likelihoods in (30) are computed from

$$\ell_{\text{acc}}^{(i)} = -\frac{1}{2} \sum_{j=1}^3 (\mathcal{Z}_{\text{acc},j}^{(i)})^2, \quad (31)$$

$$\ell_{\text{mag}}^{(i)} = -\frac{1}{2} \sum_{j=1}^3 (\mathcal{Z}_{\text{mag},j}^{(i)})^2. \quad (32)$$

Note that filter performance can be sensitive to these thresholds and may require tuning for different applications.

Once the particle diversity becomes too low and we start seeing significant weight degeneracy, indicated by $\text{ESS} < 0.5N_p$, we resample. Resampling means generating a new set of more diverse particles to replace the old degenerate ensemble. Kuptamettee et al. (2022) summarized and compared various resampling techniques for particle filters [60]. We use systematic resampling,

which draws a single initial uniform random sample and generates the remaining particles at fixed intervals over equally divided segments of $[0, 1]$, resulting in improved computational efficiency and reduced variance. After resampling, we apply roughening by adding Gaussian noise to the particle state [61]

$$\mathbf{x}_i \leftarrow \mathbf{x}_i + \mathcal{N}(\mathbf{0}, c^2 \sigma_x^2 \mathbf{I}), \quad (33)$$

$$\mathbf{v}_i \leftarrow \mathbf{v}_i + \mathcal{N}(\mathbf{0}, c^2 \sigma_v^2 \mathbf{I}), \quad (34)$$

to restore particle diversity and prevent sample impoverishment. We use $c = 0.15 N_p^{-1/6}$, $\sigma_x = 1.0 \text{ m}$, and $\sigma_v = 0.5 \text{ m s}^{-1}$ for all experiments without further tuning as these values produce robust and accurate results. Table 2 summarizes all particle filter parameters used in this study.

Parameter	Symbol	Value
Number of particles	N_p	500
Initial particle spread	σ_{init}	0.05 m, 0.05 m s ⁻¹
Fusion weight	γ	0.5
ESS resampling threshold	$0.5 N_p$	250
Maximum tempering factor	τ_{max}	64
Roughening position scale	σ_x	1.0 m
Roughening velocity scale	σ_v	0.5 m s ⁻¹
Substep factor	f	1

Table 2: Parameters used in the particle filter. Note that the particles in the filter correspond to points in state space indicating hypotheses about the state of the in-silico particle, but are not particles in the same sense.

2.6. Unscented Kalman filter

As a third filter we use an unscented Kalman filter (UKF) [62]. We keep the state $\mathbf{S}_k = [\mathbf{x}_k \ \mathbf{v}_k \ \mathbf{a}_k]^T$, the state transition function (9) and the observation operator (12) identical to those of the EKF in §2.4, so that differences in tracking performance are attributable to the filter alone.

We use the `AdditiveUnscentedKalmanFilter` class of the `pykalman` library with its default sigma point scaling and the same initial state, initial covariance $\mathbf{P}_0$, process noise covariance $\mathbf{Q}$ (11) and measurement noise covariance $\mathbf{R}_k$ (14) as the EKF. Since the flow field, and hence the transition function, depends explicitly on time, the filter is advanced one step at a time

Parameter	Symbol	Value
Acceleration measurement noise covariance	$\sigma_{a_m}^2$	0.04
Magnetic measurement noise covariance	$\sigma_{B_m}^2$	0.04
Dynamic model acceleration variance	σ_a^2	0.8

Table 3: Parameters of the noise added to the synthetic data fed to the three filters.

using the `filter_update` method with the transition function evaluated at the current time. The UKF needs positive-definite covariance matrices to calculate Cholesky decompositions [62]. We therefore regularize all covariance matrices and shift their diagonal by a small positive shift on the diagonal of 10^{-6} . No further tuning is performed.

3. Numerical results

This section presents the results of tracking in-silico particles in the two flow fields, obtained using the EKF, PF and UKF.

The choice of initial covariance matrix ($\mathbf{P}_{0,0}$) is based on the confidence we have in the initial state. We use the value $\mathbf{P}_{0,0} = 0.1\mathbf{I}_{9 \times 9}$ for both vortex and experimental flows, reflecting moderate uncertainty in the initial state estimate. The measurement noise covariances ($\sigma_{a_m}^2$ and $\sigma_{B_m}^2$) provided to the filters are intentionally set to values different from the true noise levels, reflecting a realistic scenario where the exact sensor noise characteristics are not perfectly known. Finally, the value for the acceleration variance of the dynamic model σ_a^2 was selected to minimise the tracking error using the same trajectories for which accuracy is reported below. The values of the parameters used for the filters are summarized in Table 3.

Ground-truth trajectories and synthetic data are obtained by solving (2) with high accuracy with zero initial relative velocity. A 5% Gaussian noise is added to all measurements before they are given to the filters. To assess the accuracy of the reconstructed trajectories we use the relative error

$$\text{RelErr}(t_k) = \frac{\|\hat{\mathbf{x}}(t_k) - \mathbf{x}(t_k)\|_2}{L}, \quad (35)$$

where $\|\cdot\|_2$ denotes the Euclidean norm, $\hat{\mathbf{x}}(t_k)$ and $\mathbf{x}(t_k)$ are the estimated and true particle positions at time t_k , and L is the total arc length of the true trajectory.

The magnetic field source is placed at $(0, 0, 0.3)$ for both the analytical vortex and experimental cases. In the analytical case, all quantities are treated as dimensionless from the outset, so the magnetic field model is evaluated directly at the nondimensional particle positions. However, for the experimental case, the particle position is first redimensionalized by the characteristic length L before evaluating the dipole field model (1), and the resulting field is subsequently nondimensionalized using the Earth's magnetic field as characteristic field strength $B_c = 2.5 \times 10^{-5}$ T. Therefore, the corresponding measurement Jacobian is rescaled by a factor of L/B_c .

3.1. Tracking in-silico particles in the vortex

Figure 3 shows the trajectories in the flow field (5) reconstructed by the EKF and PF over the time interval $t \in [0, 5]$. The true initial position of the particle is $(1, 0, 0)$ while the initial position state for the filters is $(1.2, 0.2, -0.1)$. Both true and estimated initial relative velocity is set to zero. For the EKF's and UKF, where the state includes acceleration, the initial acceleration is set to zero.

Figure 4 shows the relative Euclidean error (35) over time for the EKF, PF and UKF (Pykalman). The EKF and PF achieve comparable accuracy and maintain a relative error below 1% after the initial transient. By contrast, the UKF diverges progressively after $t \approx 2$ and produces errors more than three times larger by time $t = 5$. All three filters show an initial spike around $t = 0$, because of the discrepancy between the true and assumed initial positions but can correct this as more and more sensor readings arrive. It was also observed that the UKF is considerably more sensitive to the choice of the initial state covariance $\mathbf{P}_{0,0}$ than the EKF.

3.2. Tracking in-silico particles in the lab-scale STR

In the experiment, the velocity field was measured over a duration of 2 seconds and we track the in-silico particle over the full time interval. Since MaRGE is solved numerically in nondimensional form, the experimental data must also be nondimensionalized. This is done by scaling the velocity field with the characteristic velocity U and the time domain $[0, 2]$ by the characteristic time $T = L/U$. As a result, the trajectories obtained using MaRGE, as well as the EKF and PF estimates, are expressed in nondimensional form. However, to allow comparison of the results shown here to the experimental setup, all shown quantities have been redimensionalized by multiplication with their respective reference. The initial position of the particle is set to

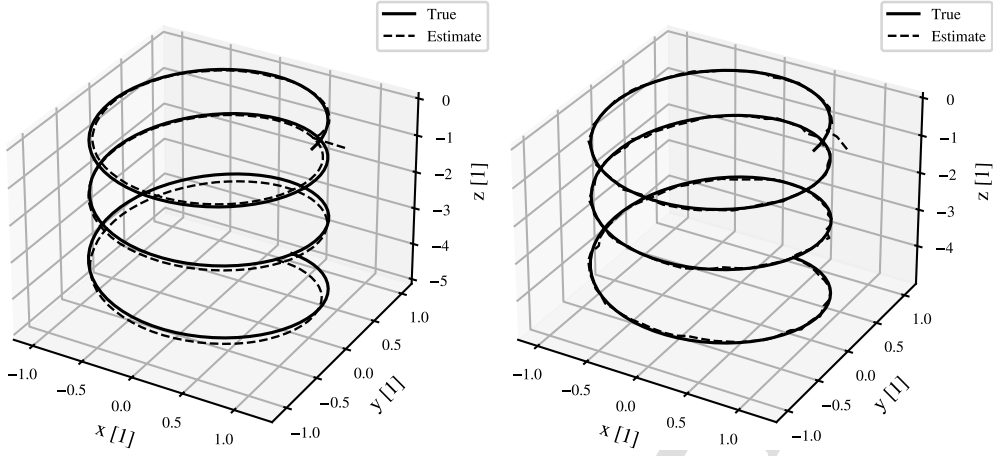


Figure 3: Reconstructed trajectories over the time interval $[0, 5]$ in the three-dimensional vortex. The solid line represents the ground-truth while the dashed line shows the estimates provided by the EKF (left) and the PF (right).

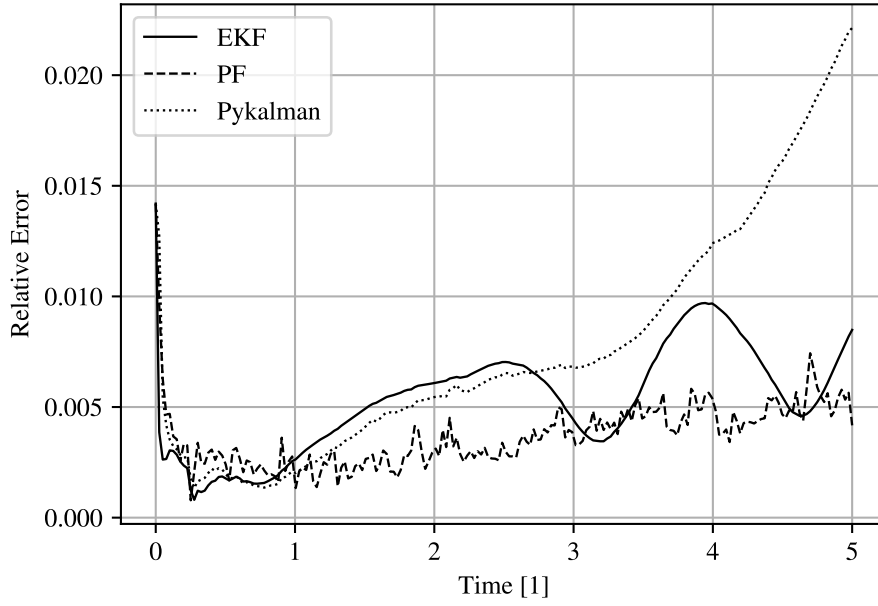


Figure 4: Relative error (35) over time for the EKF, PF and UKF (Pykalman) for an in-silico LS in the three-dimensional vortex.

(0, 0, 0.21) m while the initial position estimate for both filters differs from the true initial position by approximately 1%.

Figure 5 shows the relative Euclidean error over time for the EKF and PF in the experimental flow field. The EKF produces smaller error than the PF throughout most of the tracking interval, although it produces a more accurate estimate at the very end. Errors for both filters are below 6% throughout the entire interval with averages around 1.1% for the EKF and 2.2% for the PF. Due to the filter's stochastic nature, the PF error is considerably more volatile. This [volatility](#) suggests that the particle ensemble struggles to represent the posterior distribution accurately in the more complex, experimentally obtained flow field. Both filters show a small initial transient near $t = 0$, consistent with the slight offset in the initial position estimate.

UKF results are not shown here since, with $\mathbf{P}_{0,0} = 0.1 \mathbf{I}_9$, it produced a mean relative error exceeding 1800%. Reducing the initial state covariance to $\mathbf{P}_{0,0} = 10^{-2} \mathbf{I}_9$ or less, while keeping the flow field and all other parameters unchanged, allows tracking of particles again with a mean relative error of 1.6%, comparable to the 1.1 and 2.2 obtained with the EKF and PF. The EKF and PF, by contrast, are insensitive to this parameter across the same range. With the characteristic length $L = 0.13$ m, $\mathbf{P}_{0,0} = 0.1 \mathbf{I}_9$ corresponds to a prior standard deviation of 41 mm per component in a vessel of 130 mm diameter and 230 mm height, reflecting a moderate degree of uncertainty about the initial position of a Lagrangian sensor that is realistic in operation. The values at which the UKF succeeds correspond to 13 mm and 4 mm and would require the initial position to be known to within 10% and 3% of the vessel diameter. This level of precision will be difficult to achieve in practice. The relevant conclusion is therefore not that the UKF cannot solve the tracking problem, but that the range of initial covariances it tolerates is substantially narrower than that of the EKF and PF and that it fails once that range is left.

4. Conclusions

The three-dimensional tracking of Lagrangian sensors in chemical reactors is challenging due to the complex dynamics. Our paper proposes a tracking framework that combines filtering algorithms (EKF, UKF and PF) using physics-based models with sensor readings from an accelerometer and magnetometer measuring an artificial magnetic field. The framework is tested on

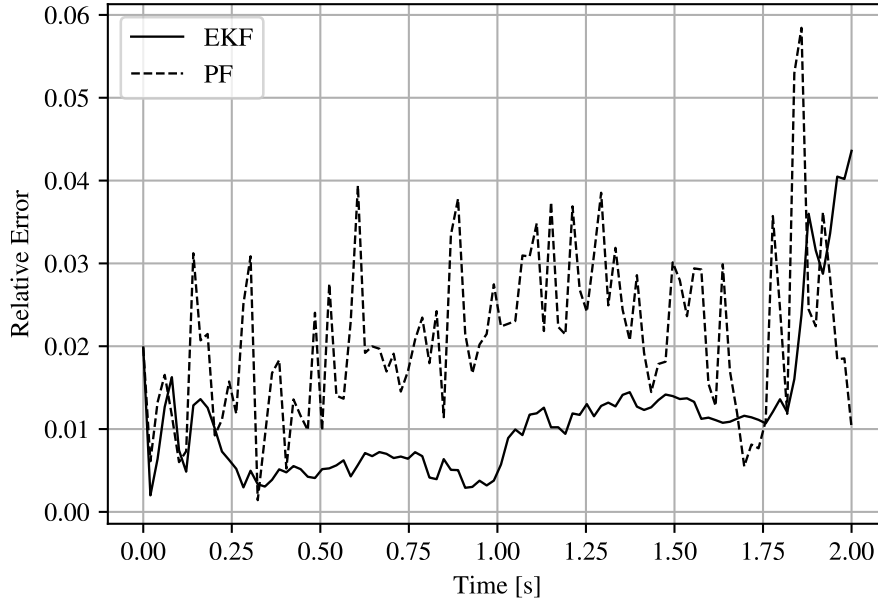


Figure 5: Relative Euclidean error (35) over time for the extended Kalman filter and particle filter (PF) for an in-silico LS in the lab-scale STR.

in-silico replicas of the sensors recently proposed by Geläsčius et al. [37], first in an idealized three-dimensional vortex and then in a lab-scale stirred tank reactors, where the flow-field was characterized by particle image velocimetry.

We show that EKF and PF can reliably reconstruct trajectories of in-silico particles with relative errors of the order of a few percent, even when faced with noisy data and an imperfect process model. *By contrast, the generic UKF provided by thepykalman library requires an initial state covariance an order of magnitude smaller than the EKF and PF to successfully track the particle in the stirred tank reactor. Such a small initial covariance corresponds to a degree of prior knowledge about the sensor position that is not realistically available.* While substantial challenges remain, see the discussion of limitations and next steps below, the proposed framework shows promise to eventually allow online three-dimensional tracking of Lagrangian sensors.

Limitations and future work. While our study demonstrates the promise of combining IMU data with bespoke filters to enable full 3D tracking of La-

grangian sensors in chemical reactors, important challenges remain that need to be addressed before the approach could be used operationally. Most importantly, while the flow field in the lab-scale reactor was measured experimentally, the reconstructed trajectories are simulated, even though neglecting the Basset force in the filters introduces a substantial modelling error that both EKF and PF were able to cope with. The Maxey-Riley-Gatignol equations (MaRGE) are generally accepted as a useful model for the dynamics of inertial particles [22, §5.3.1]. However, to date there is no analysis of how well MaRGE describes the movement of Lagrangian sensors, which are relatively large inertial particles and where the geometric center and the center of mass might not be perfectly aligned. Therefore, we do not know the gap between the dynamic models used in the filters and the motion of real-world LS with high certainty. Reconstructing experimentally measured trajectories of real Lagrangian particles in the lab-scale reactor is planned for future work.

Furthermore, our proposed setup, combining readings from an accelerometer and magnetometer for tracking using an externally generated magnetic field, can only be used in reactors made out of non-magnetic materials. In steel vessels, for example, other sensors would have to be employed to counteract the drift from dead reckoning. Pressure sensors have been used for positioning particles in the vertical [12] and could also be integrated into our filters for improved vertical positioning. Other options might be tracking based on ultrasound [63]. While our filter algorithms can be adopted to include additional or different sensor modalities, [such modifications](#) would influence tracking performance and thus require further tests and investigation.

In our tracking experiments, we assumed that sensor noise is additive Gaussian noise but the noise characteristics for real particles, having to transmit their IMU readings to a processing unit outside of the reactor via low-power communication, might be quite different. However, modern machine learning architectures like autoencoders have proven to be quite effective at denoising sensor data and could be a useful remedy [64]. Therefore, the issue of sensor and communication noise can likely be dealt with satisfactorily.

Lastly, the tracking algorithms in this study were run offline and cannot yet track particles in real-time. Enabling online tracking will require to speed up computations by about two orders of magnitude. This would require to translate the current Python code into a more efficient compiled programming language like C/C++ and to make effective use of compiler optimization as well as the use parallel processing on multi-core CPUs or

graphics processing units (GPUs). While [such a rewrite](#) would require a substantial investment in terms of development time, the necessary techniques are well tested and readily available.

References

- [1] A. Amador, M. Canals, Design and development of an instrumented drifter for lagrangian measurements of inertial particle dynamics in breaking waves, *IEEE Journal of Oceanic Engineering* 41 (1) (2016) 82–93. doi:10.1109/JOE.2015.2389591.
URL <https://dx.doi.org/10.1109/JOE.2015.2389591>
- [2] F. Feddersen, A. Amador, K. Pick, A. Vizuet, K. Quinn, E. Wolfinger, J. H. MacMahan, A. Fincham, The wavedrifter: a low-cost imu-based lagrangian drifter to observe steepening and overturning of surface gravity waves and the transition to turbulence, *Coastal Engineering Journal* 66 (1) (2024) 44–57. doi:10.1080/21664250.2023.2238949.
URL <https://doi.org/10.1080/21664250.2023.2238949>
- [3] R. Cox, S. Felder, Injury-free transport of fish through closed conduit components, *Journal of Ecohydraulics* 11 (1) (2026) 1–16.
- [4] G. Iddan, G. Meron, A. Glukhovsky, P. Swain, Wireless capsule endoscopy, *Nature* 405 (6785) (2000) 417–417.
- [5] K. Kalantar-Zadeh, K. J. Berean, N. Ha, A. F. Chrimes, K. Xu, D. Grando, J. Z. Ou, N. Pillai, J. L. Campbell, R. Brkljača, et al., A human pilot trial of ingestible electronic capsules capable of sensing different gases in the gut, *Nature Electronics* 1 (1) (2018) 79–87.
- [6] J. Bryant, Ph.D. thesis, University of Cambridge (1969).
- [7] R. Mann, P. Mavros, J. Middleton, A structured stochastic flow model for interpreting flow-follower data from a stirred vessel, *Trans IChemE* 59 (1981) 271–278.
- [8] J. Bisgaard, M. Muldbak, T. Tajsleiman, T. Rydal, T. Rasmussen, J. K. Huusom, K. V. Gernaey, Characterization of mixing performance in bioreactors using flow-following sensor devices, Ph.D. thesis (2021).

- [9] J. Bisgaard, M. Muldbak, S. Cornelissen, T. Tajssoleiman, J. K. Huusom, T. Rasmussen, K. V. Gernaey, Flow-following sensor devices: A tool for bridging data and model predictions in large-scale fermentations, *Computational and Structural Biotechnology Journal* 18 (2020) 2908–2919. doi:10.1016/j.csbj.2020.10.004. URL <https://doi.org/10.1016/j.csbj.2020.10.004>
- [10] M. Muldbak, C. Gargalo, U. Krühne, I. Udugama, K. V. Gernaey, Digital twin of a pilot-scale bio-production setup, in: Y. Yamashita, M. Kano (Eds.), 14th International Symposium on Process Systems Engineering, Vol. 49 of Computer Aided Chemical Engineering, Elsevier, 2022, p. 1417–1422. doi:10.1016/B978-0-323-85159-6.50236-0. URL <https://doi.org/10.1016/B978-0-323-85159-6.50236-0>
- [11] C. Haringa, T. Tajssoleiman, W. A. van Winden, D. Dong, R. M. Gladue, L. Wu, T. Rasmussen, H. J. Noorman, Flow-following sensor technology, a route to validated cfd models, *Biochemical Engineering Journal* 215 (2025) 109623. doi:10.1016/j.bej.2024.109623. URL <https://doi.org/10.1016/j.bej.2024.109623>
- [12] J. Bisgaard, T. Tajssoleiman, M. Muldbak, T. Rydal, T. Rasmussen, J. K. Huusom, K. V. Gernaey, Automated compartment model development based on data from flow-following sensor devices, *Processes* 9 (9) (2021). doi:10.3390/pr9091651. URL <https://www.mdpi.com/2227-9717/9/9/1651>
- [13] J. Bisgaard, M. Muldbak, T. Tajssoleiman, T. Rydal, T. Rasmussen, J. K. Huusom, K. V. Gernaey, Characterization of mixing performance in bioreactors using flow-following sensor devices, *Chemical Engineering Research and Design* 174 (2021) 471–485. doi:10.1016/j.cherd.2021.08.008. URL <https://doi.org/10.1016/j.cherd.2021.08.008>
- [14] S. F. Reinecke, L. Buntkiel, R. Kipping, U. Hampel, Process characterization in industrial vessels by flow-following sensor particles, *Measurement Science and Technology* 33 (9) (2022). doi:10.1088/1361-6501/ac75af. URL <https://doi.org/10.1088/1361-6501/ac75af>

- [15] S. Hofmann, L. Buntkiel, R. Rautenbach, L. Gaugler, Y. Ma, I. Haase, J. Fitschen, T. Wucherpennig, S. F. Reinecke, M. Hoffmann, R. Takors, U. Hampel, M. Schlüter, Experimental analysis of lifelines in a 15,000 l bioreactor by means of lagrangian sensor particles, *Chemical Engineering Research and Design* 205 (2024) 695–712. doi:10.1016/j.cherd.2024.04.015.
URL <https://doi.org/10.1016/j.cherd.2024.04.015>
- [16] S. Hofmann, R. Rautenbach, L. Buntkiel, I. S. Brouwers, L. Gaugler, J. Barczyk, J. Fitschen, S. Reinecke, M. Hoffmann, R. Takors, et al., Lagrangian sensor particles for detecting hydrodynamic heterogeneities in industrial bioreactors: experimental analysis and lattice-boltzmann simulations, *Chemical Engineering Journal Advances* 22 (2025) 100744.
- [17] S. Reinecke, A. Deutschmann, K. Jobst, H. Kryk, E. Friedrich, U. Hampel, Flow following sensor particles—validation and macro-mixing analysis in a stirred fermentation vessel with a highly viscous substrate, *Biochemical engineering journal* 69 (2012) 159–171.
- [18] S. F. Reinecke, A. Deutschmann, K. Jobst, U. Hampel, Macro-mixing characterisation of a stirred model fermenter of non-newtonian liquid by flow following sensor particles and ert, *Chemical Engineering Research and Design* 118 (2017) 1–11.
- [19] R. Rautenbach, S. Hofmann, L. Buntkiel, J. Schäfer, S. F. Reinecke, M. Hoffmann, U. Hampel, M. Schlüter, Dynamics of lagrangian sensor particles: The effect of non-homogeneous mass distribution, *Processes* 12 (8) (2024). doi:10.3390/pr12081617.
URL <https://dx.doi.org/10.3390/pr12081617>
- [20] J. A. Sykes, D. Weston, N. Adio, H. Che, R. Hart-Villamil, A. L. Nicuşan, W. Peace, D. Rhymer, D. Werner, T. Kokalova-Wheldon, A. Ingram, C. R. Windows-Yule, Validation of simulations of particulate, fluid and multiphase systems using positron emission particle tracking: A review, *Particuology* 101 (2025) 117–145. doi:10.1016/j.partic.2024.06.015.
URL <https://doi.org/10.1016/j.partic.2024.06.015>
- [21] S. Hofmann, C. Weiland, J. Fitschen, A. von Kameke, M. Hoffmann, M. Schlüter, Lagrangian sensors in a stirred tank reactor: Comparing

- trajectories from 4d-particle tracking velocimetry and lattice-boltzmann simulations, *Chemical Engineering Journal* 449 (2022) 137549. doi:10.1016/j.cej.2022.137549.
URL <https://doi.org/10.1016/j.cej.2022.137549>
- [22] C. Tropea, A. L. Yarin, J. F. Foss, et al., *Springer handbook of experimental fluid mechanics*, Vol. 1, Springer, 2007.
- [23] L. Buntkiel, Y. Ma, S. F. Reinecke, U. Hampel, Orientation resolved measurements of accelerations with sensor particles in bioreactors, *tm - Technisches Messen* 90 (12) (2023) 835–845. doi:10.1515/teme-2023-0115.
URL <https://doi.org/10.1515/teme-2023-0115>
- [24] S. Lange, Reliable localization in a bioreactor, *Microelectronic News*, accessed February 2026 (2019).
- [25] J. Cox, W. Zhang, T. Furukawa, Motion tracking with coupled magnetometers and dynamic imu measurement fusion in nonuniform magnetic fields, in: 2023 26th International Conference on Information Fusion (FUSION), 2023, p. 1–8. doi:10.23919/FUSION52260.2023.10224146.
URL <https://doi.org/10.23919/FUSION52260.2023.10224146>
- [26] H. Hellmers, A. Norrdine, J. Blankenbach, A. Eichhorn, An imu/magnetometer-based indoor positioning system using kalman filtering, in: *International Conference on Indoor Positioning and Indoor Navigation*, 2013, p. 1–9. doi:10.1109/IPIN.2013.6817887.
URL <https://dx.doi.org/10.1109/IPIN.2013.6817887>
- [27] M. Menolotto, D.-S. Komaris, S. Tedesco, B. O’Flynn, M. Walsh, Motion capture technology in industrial applications: A systematic review, *Sensors* 20 (19) (2020). doi:10.3390/s20195687.
URL <https://www.mdpi.com/1424-8220/20/19/5687>
- [28] F. Wu, Y. Liang, Y. Fu, X. Ji, A robust indoor positioning system based on encoded magnetic field and low-cost imu, in: *2016 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, IEEE, 2016, pp. 204–212.

- [29] D. Duckworth, pykalman: Kalman filter, smoother, and em algorithm for python, <https://github.com/pykalman/pykalman>, accessed: YYYY-MM-DD (2012).
- [30] V. Rathi, Vamika-rathi/in-silico-tracking: In-silico lagrangian sensor tracking in a stirred tank reactor: filtering code (Jun. 2026). doi: 10.5281/zenodo.20629013. URL <https://doi.org/10.5281/zenodo.20629013>
- [31] D. Luenberger, Observers for multivariable systems, IEEE transactions on automatic control 11 (2) (1966) 190–197.
- [32] P. Bernard, V. Andrieu, D. Astolfi, Observer design for continuous-time dynamical systems, Annual Reviews in Control 53 (2022) 224–248. doi:10.1016/j.arcontrol.2021.11.002.
- [33] C. Kravaris, J. Hahn, Y. Chu, Advances and selected recent developments in state and parameter estimation, Computers & chemical engineering 51 (2013) 111–123.
- [34] J. B. Rawlings, B. R. Bakshi, Particle filtering and moving horizon estimation, Computers & chemical engineering 30 (10-12) (2006) 1529–1541.
- [35] M. Zeitz, The extended luenberger observer for nonlinear systems, Systems & Control Letters 9 (2) (1987) 149–156.
- [36] V. Pasku, A. De Angelis, G. De Angelis, D. D. Arumugam, M. Dionigi, P. Carbone, A. Moschitta, D. S. Ricketts, Magnetic field-based positioning systems, IEEE Communications Surveys & Tutorials 19 (3) (2017) 2003–2017. doi:10.1109/COMST.2017.2684087.
- [37] A. Geläsčius, L. Rennpferdt, V. Rathi, M. Hussain, D. Ruprecht, H. K. Trieu, Millimeter-scale lagrangian sensor spheres for local state monitoring in smart reactors: Towards energy-autonomous operation, Submitted to Industrial & Engineering Chemistry Research (2026).
- [38] A. Edelstein, Advances in magnetometry, Journal of Physics: Condensed Matter 19 (16) (2007) 165217. doi:10.1088/0953-8984/19/16/165217. URL <https://dx.doi.org/10.1088/0953-8984/19/16/165217>

- [39] M. R. Maxey, J. J. Riley, Equation of motion for a small rigid sphere in a nonuniform flow, *The Physics of Fluids* 26 (4) (1983) 883–889. doi:10.1063/1.864230.
- [40] R. Gatignol, The Faxen Formulae For A Rigid Particle In An Unsteady Non-Uniform Stokes Flow, *Journal De Mecanique Theorique Et Appliquee* 2 (2) (1983) 143–160.
- [41] A. Daitche, Advection of inertial particles in the presence of the history force: Higher order numerical schemes, *Journal of Computational Physics* 254 (2013) 93–106. doi:10.1016/j.jcp.2013.07.024.
- [42] J. Urizarna-Carasa, D. Ruprecht, A. von Kameke, K. Padberg-Gehle, Relevance of the basset history term for lagrangian particle dynamics, *Chaos: An Interdisciplinary Journal of Nonlinear Science* 35 (7) (2025).
- [43] V. Rathi, D. Ruprecht, Numerical modeling of inertial particles in three-dimensional fluid flow, *PAMM* 26 (2) (2026) e70158.
- [44] S. G. Prasath, V. Vasan, R. Govindarajan, Accurate solution method for the maxey–riley equation, and the effects of basset history, *Journal of Fluid Mechanics* 868 (2019) 428–460.
- [45] C. Crowe, M. Sommerfeld, Y. Tsuji, et al., *Multiphase flows with droplets and particles*, CRC Press Taylor Francis Group, Boca Raton, 1998.
- [46] F. Candelier, J. Angilella, M. Souhar, On the effect of the boussinesq–basset force on the radial migration of a stokes particle in a vortex, *Physics of Fluids* 16 (5) (2004) 1765–1776.
- [47] C. Weiland, E. Steuwe, J. Fitschen, M. Hoffmann, M. Schlüter, K. Padberg-Gehle, A. von Kameke, Computational study of three-dimensional lagrangian transport and mixing in a stirred tank reactor, *Chemical Engineering Journal Advances* 14 (2023) 100448. doi:10.1016/j.cej.2023.100448.
URL <https://www.sciencedirect.com/science/article/pii/S2666821123000066>
- [48] D. Schanz, S. Gesemann, A. Schröder, Shake-the-box: Lagrangian particle tracking at high particle image densities, *Experiments in Fluids*

- 57 (5) (2016) 70. doi:10.1007/s00348-016-2157-1.
URL <https://doi.org/10.1007/s00348-016-2157-1>
- [49] A. Kameke, R. Uphoff, E. Steuwe, J. Nissen, M. Hoffmann, M. Schl, F. Kexel, Experimental analysis of time resolved three-dimensional velocity and vorticity fields behind single rising bubbles using lagrangian particle tracking velocimetry, *Chemical Engineering Science* (2026) 123403.
- [50] M. Baer, findiff software package, <https://github.com/maroba/findiff> (2018).
URL <https://github.com/maroba/findiff>
- [51] L. A. McGee, S. F. Schmidt, Discovery of the kalman filter as a practical tool for aerospace and industry, Tech. rep. (1985).
- [52] A. Becker, Kalman filter from the ground up, *Kalmanfilter. net*, 2024.
- [53] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, Y. Katariya, C. Leary, D. Maclaurin, G. Necoara, A. Paszke, J. VanderPlas, S. Wanderman-Milne, Q. Zhang, JAX: composable transformations of Python+NumPy programs (2018).
URL <http://github.com/jax-ml/jax>
- [54] P. Djuric, J. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. Bugallo, J. Miguez, Particle filtering, *IEEE Signal Processing Magazine* 20 (5) (2003) 19–38. doi:10.1109/MSP.2003.1236770.
- [55] Z. Chen, Bayesian filtering: From kalman filters to particle filters, and beyond, *Statistics* 182 (01 2003). doi:10.1080/02331880309257.
- [56] J. Elfring, E. Torta, R. van de Molengraft, Particle filters: A hands-on tutorial, *Sensors (Basel, Switzerland)* 21 (2021).
URL <https://api.semanticscholar.org/CorpusID:231595612>
- [57] B. Siebler, O. Heirich, A. Lehner, S. Sand, U. Hanebeck, Robust particle filter for magnetic field-based train localization, 2022. doi:10.33012/2022.18536.

- [58] H. Zhang, F. Martin, Robotic mapping assisted by local magnetic field anomalies, in: 2011 IEEE Conference on Technologies for Practical Robot Applications, 2011, pp. 25–30. doi:10.1109/TEPRA.2011.5753476.
- [59] E. Herbst, F. Schorfheide, Tempered particle filtering, *Journal of Econometrics* 210 (1) (2019) 26–44.
- [60] C. Kuptamettee, N. Aunsri, A review of resampling techniques in particle filtering framework, *Measurement* 193 (2022) 110836.
- [61] T. Li, S. Sun, T. P. Sattar, J. M. Corchado, Fight sample degeneracy and impoverishment in particle filters: A review of intelligent approaches, *Expert Systems with applications* 41 (8) (2014) 3944–3954.
- [62] S. J. Julier, J. K. Uhlmann, New extension of the kalman filter to nonlinear systems, in: *Signal processing, sensor fusion, and target recognition VI*, Vol. 3068, Spie, 1997, pp. 182–193.
- [63] Q. Chen, F.-W. Liu, S. K. Cho, K. Kim, 3-d real-time ultrasound tracking of acoustically actuated swimming microdrone, *Scientific Reports* 14 (1) (2024) 1547.
- [64] K. Berahmand, F. Daneshfar, E. S. Salehi, Y. Li, Y. Xu, Autoencoders and their applications in machine learning: a survey, *Artificial intelligence review* 57 (2) (2024) 28.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: