2020 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 21–24, 2020, ESPOO, FINLAND

SIMULTANEOUS INTENT PREDICTION AND STATE ESTIMATION USING AN INTENT-DRIVEN INTRINSIC COORDINATE MODEL

Jiaming Liang, Bashar I. Ahmad, Simon Godsill

Signal Processing and Communications (SigProC) Laboratory
Engineering Department, University of Cambridge, Cambridge, U.K. CB2 1PZ
Email: {jl809, bia23, sjg30}@cam.ac.uk

ABSTRACT

The motion of an object (e.g. ship, jet, pedestrian, bird, drone, etc.) is usually governed by premeditated actions as per an underlying intent, for instance reaching a destination. In this paper, we introduce a novel intent-driven dynamical model based on a continuous-time intrinsic coordinate model. By combining this model with particle filtering, a seamless approach for jointly predicting the destination and estimating the state of a highly manoeuvrable object is developed. We examine the proposed inference technique using real data with different measurement models to demonstrate its efficacy. In particular, we show that the introduced approach can be a flexible and competitive alternative, in terms of prediction and estimation performance, to other existing methods for various measurement models including nonlinear ones.

Index Terms— tracking algorithms, intent prediction, particle filters, variable rate models, intrinsic coordinate model

1. INTRODUCTION

Numerous tracking algorithms have been introduced for estimating hidden quantities related to an object (e.g. its position, higher order kinematics or other spatial-temporal characteristics) from noisy observations [1]. While the primary objective of the majority of these methods is the accurate estimation of the object kinematic state, in this paper we focus on devising a reliable solution for inferring an object's intended destination $\mathcal{D}$, out of a finite nominal set $\mathbb{D} = \{\mathcal{D}_i, i = 1, 2, \ldots, N_{\mathcal{D}}\}$, as well as estimating its state. This facilities informed action planning (e.g. to circumvent conflict) and have applications in various areas such as surveillance, smart navigation, automotive and others. The considered targets (e.g. a speed boat, jet, bird, pedestrian, racing drone, etc.) can undertake, albeit not necessarily, intentional fast manoeuvres such as rapid turning movements and large accelerations, for instance to hone in on $\mathcal{D}$, to avoid obstacles or due to abrupt change in intent.

Inferring complicated intentions, such as of drivers and pedestrians in complex environments, can be tackled with several data-driven methods [2–4]. They often require sufficiently complete training data (not always available) to learn a representative model. Differently, model-based approaches have been shown to be effective for the same task, while having minimal (if any) training requirements, especially for a finite set of possible intents [3, 5]. On the other hand, destination-aware tracking and Ornstein-Uhlenbeck (OU) processes can improve the inference quality by incorporating, typically *known*, predictive information such as endpoint or velocity mean [6–9]. Stochastic context-free grammar and reciprocal chain models were also proposed to integrate-predict destination $\mathcal{D}$ [10–13]. In this paper, the destination is unknown *a priori* and a novel model-based approach, which does not require discretisation of the state space and/or constraining the shape of the target trajectory to $\mathcal{D}$ as in [10–12] and [14], is introduced to jointly estimate the intent and (kinematic) state of a highly manoeuvrable object.

Previous work on intent prediction in object tracking using continuous-time state space models, e.g. [5, 15, 16], exclusively considered linear and Gaussian dynamic systems. This led to Kalman filter-based algorithms where intent prediction and state estimation are treated separately. In [17] a mean-reverting jump diffusion process is proposed to model sharp unintentional movements due to external forces. It focuses on capturing perturbations-induced movements, jolts, and utilises Rao-Blackwellisation which may not always be applicable with complex (e.g. nonlinear and/or non-Gaussian) measurement models. On the contrary, a more elaborate (nonlinear) dynamical model, which can naturally describe a sequence of deliberate and persistent destination-driven manoeuvres, is introduced in this paper. An inference approach to the simultaneous intent and state estimation, based on the proposed intent-driven model, is also presented. It uses a variable rate particle filter [18] and is generally applicable to various sensor measurement models, including nonlinear and/or non-Gaussian ones. It should be emphasised that the intent inference procedure in this paper is fundamentally distinct from that based on the sequential evaluation of model evidence (e.g. in [5, 15–17]), since the endpoint here is part of the system latent state.

The main contribution of this paper is the development of a novel intent-driven intrinsic model (IDIM), which can capture the influence of destination on the motion of agile objects. This model bears a resemblance to the variable rate intrinsic coordinate model in [18, 19], hence it is suitable for objects that can perform swift manoeuvres. IDIM is also a continuous-time model and can naturally handle asynchronous sensor measurements. In terms of intent modelling, it is closely related to interacting multiple model (IMM)-based particle filtering methods [20, 21], where a mode variable is used to govern the transition among a set of candidate models. However, here we employ this mode variable to guide the changes in intent, whereas IMMs utilise it to account for different motion patterns. Nonetheless, the latter aspect is seamlessly handled by the intrinsic motion model. In addition, we introduce a kinetic state dependent transition density for the mode variable, which facilitates autonomous transitions in destination, in comparison to conventional multiple model techniques in which a constant Markov chain transition matrix is often used.

The proposed intent-driven intrinsic model is generic where various designs of the force directing the object to its destination can

978-1-7281-6662-9/20/$31.00 ©2020 IEEE

be incorporated, see Section 2. Consequently, IDIM can describe sophisticated intent-directed motion while retaining its ability to accommodate varied degrees of manoeuvrability. Additionally, effective inference with this model can be accomplished by the use of a variable rate particle filter, thereby allowing for the joint sequential estimation of the object intent and kinematics. Real data from racing drones is used to demonstrate the competitive performance of the introduced approach in challenging on-line tracking scenarios, compared with other considered methods.

The rest of the paper is structured as follows. Section 2 gives a detailed description of the motion and intent modelling. The inference algorithm associated with the proposed IDIM is outlined in Section 3. We show in Section 4 the experimental results from real data before we draw conclusions in Section 5.

2. PROPOSED MODELLING

2.1. Intrinsic Coordinate Motion Model

While various nonlinear mathematical models exist for manoeuvring objects [18, 22], we build our model here based upon a continuous-time coordinate-coupled manoeuvring model, known as intrinsic coordinate model or curvilinear motion model [18, 19, 22, 23]. This class of models is realistic in that it describes motion of targets under self-forcing (e.g. with on-board propelling mechanism); it is formulated using forces applied relative to the body frame of the object; and the forces are permitted to arrive stochastically and asynchronously with the measurements, resembling real-world piloting behaviours. In this work, the object is considered as a particle instead of an extended one. The presentation of the intrinsic coordinate model given below mainly follows that of [18]. The governing equations for the model are given by

$$T_{T,k} = \lambda \frac{ds}{dt} + m\frac{d^2 s}{dt^2}, \tag{1}$$

$$T_{P,k} = m\frac{ds}{dt}\frac{d\psi}{dt}, \qquad t \in (\tau_k, \tau_{k+1}], \tag{2}$$

with s being the distance a particle has travelled, ψ the heading angle with respect to a specified reference axis in the Cartesian coordinate frame (normally x), $T_{T,k}$ the tangential force, $T_{P,k}$ the force perpendicular to $T_{T,k}$, λ the friction coefficient and m the particle mass. The forces $T_{T,k}$ and $T_{P,k}$ are assumed to remain constant in the time interval $(\tau_k, \tau_{k+1}]$. This leads to deterministic evolution functions for the related states in the same interval, as will be shown below. It should be stressed that the model considered here features a variable rate design, meaning that state arrival time τ_k does not have to be synchronised with observation arrival time (assuming $N+1$ observations are made at times $\{t_0, t_1, \ldots, t_N\}$). This setting renders the model different from the conventional Markovian state space model, although the latter can be readily obtained by making $\tau_k - \tau_{k+1}$ tied to the measurement sampling interval.

An expression for speed $v = \frac{ds}{dt}$ may be obtained by solving (1) directly:

$$v(\tau_k + \Delta\tau) = v(\tau_k)e^{-\frac{\Delta\tau\lambda}{m}} + \frac{T_{T,k}}{\lambda}\left(1 - e^{-\frac{\Delta\tau\lambda}{m}}\right). \tag{3}$$

The distance s can be routinely obtained by further integrating (3). Now, (2) can be solved to give the updating expression of heading angle, based on the result for speed,

$$\psi(\tau_k + \Delta\tau) = \psi(\tau_k) + \frac{T_{P,k}}{T_{T,k}}\left(\frac{\Delta\tau\lambda}{m} - \log\left|\frac{v(\tau_k)}{v(\tau_k + \Delta\tau)}\right|\right). \tag{4}$$

Note that a closed form solution regarding the change in Cartesian position will not be available unless the coefficient λ is set to 0. Alternatively, we may obtain the 2-D Cartesian position, $\mathbf{p}_n = [x_n,\ y_n]^T$, at observation time t_n by Euler approximation,

$$\mathbf{p}(t + \delta t) \approx \mathbf{p}(t) + \delta t \cdot v(t) \cdot \begin{bmatrix}\cos\psi(t)\\ \sin\psi(t)\end{bmatrix}. \tag{5}$$

Equations (3)-(5) depict deterministic transitions for states between time τ_k and $\tau_k + \Delta\tau$, provided $T_{T,k}$, $T_{P,k}$ and relevant state values at τ_k. The state arrival time can be also interpreted as the changepoint occurrence time, since the variable rate state process is closely related to Marked Point Processes [24]. The stochasticity in the dynamic system is introduced via the changepoint time τ_k and the applied forces, i.e. $T_{T,k}$ and $T_{P,k}$. Without loss of generality, here we adopt the gamma distribution for the inter-arrival time between changepoints

$$\tau_k - \tau_{k-1} \sim \mathrm{Gam}(\alpha, \beta), \tag{6}$$

whereas for the applied forces T_T and T_P Gaussian distributions $T_k \sim \mathcal{N}(\mu, \sigma^2)$ are considered. The mean μ is the key leading to the development of intent-driven intrinsic model, which will be illustrated in details in Section 2.2.

2.2. Modelling the Intent

The intent of an object that is moving towards some destination can be revealed by a series of manoeuvres it performs over the observation period. These manoeuvres will have a direct impact on the accelerations the target undertakes. The fact that the intrinsic dynamic model is characterised by the forces applied on the object (manoeuvre actions) as well as the τ's (times defining manoeuvre periods) makes the model a promising candidate for object intent modelling.

The basic idea is to have the resultant force pointed towards the destination. Recalling that in an intrinsic coordinate system, T_T and T_P can be viewed as the forces applied on the tangential and perpendicular unit vectors that define an object's local frame in 2-D. Denoting the tangential unit vector as $\hat{\mathbf{e}}_T$, we define $\hat{\mathbf{e}}_T = \frac{\mathbf{v}}{\|\mathbf{v}\|}$ where $\mathbf{v}$ is the velocity vector. As a result, the unit vector $\hat{\mathbf{e}}_P$ perpendicular to $\hat{\mathbf{e}}_T$ can be readily obtained. A schematic representation of the intent-driven intrinsic model is given in the Figure 1. Note we have assumed the intrinsic coordinate system is right-handed. Specifically, at time τ_k^+ (timing immediately after τ_k) a force $\mathbf{f}_\mathcal{D}$ acting along the line connecting the target and the destination is given by

$$\mathbf{f}_{\mathcal{D},k} = F \times \frac{\mathbf{p}_\mathcal{D} - \mathbf{p}_k}{\|\mathbf{p}_\mathcal{D} - \mathbf{p}_k\|}$$

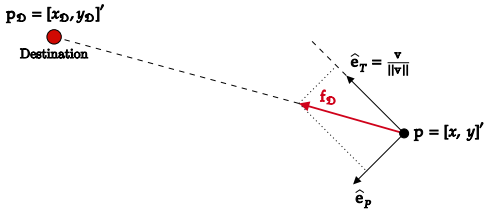

Fig. 1: Intent-driven intrinsic model; $\hat{\mathbf{e}}_T$ tangential unit vector; $\hat{\mathbf{e}}_P$ perpendicular unit vector; $\mathbf{v}$ velocity; $\mathbf{f}_\mathcal{D}$ the dragging force towards the destination; object position $\mathbf{p} = [x, y]'$; destination position $\mathbf{p}_\mathcal{D} = [x_\mathcal{D}, y_\mathcal{D}]'$.

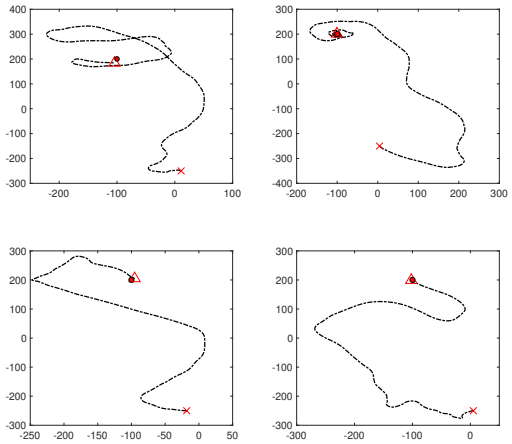

Fig. 2: Synthesised tracks: × start point; △ end point; ● destination.

with F being a value that may be state dependant and $\mathbf{p}_k$ the target position at τ_k. Let A equal some predetermined (constant) value, the following choices for F may be considered depending on the expected behaviour of the object: 1). $F = A$; 2). $F = A\phi(\cdot)$ with $\phi(\cdot)$ being some application specific function. For instance, if the object tends to decelerate as it approaches its intended destination, $\phi(\cdot)$ can be some non-increasing function taking object-destination distance $\|\mathbf{p}_{\mathcal{D}} - \mathbf{p}\|$ as its input. Now, if the unit vectors representing the target's intrinsic coordinate, i.e. $\hat{\mathbf{e}}_{T,k}$ and $\hat{\mathbf{e}}_{P,k}$, are provided, it is possible to obtain the following force components:

$$\mu_{T,k} = \mathbf{f}_{\mathcal{D},k} \cdot \hat{\mathbf{e}}_{T,k}, \quad \mu_{P,k} = \mathbf{f}_{\mathcal{D},k} \cdot \hat{\mathbf{e}}_{P,k}.$$

with "$\cdot$" being the dot product operator. Recall that the driving forces are normally distributed, now we can specify the parameters of the distributions as follows

$$T_{T,k} \sim \mathcal{N}(\mu_{T,k},\ \sigma_T^2), \quad T_{P,k} \sim \mathcal{N}(\mu_{P,k},\ \sigma_P^2) \tag{7}$$

where σ_T and σ_P are design parameters that modulate the resultant force. With this formulation, the resultant force induced by the intended endpoint will have its mean pointed to the destination. Figure 2 shows four trajectories synthesised by the proposed model.

We have shown by far how the influence of an intended destination on an object can be modelled when an intrinsic coordinate model is under consideration. In order to perform intent inference with the proposed modelling given a finite set of possible destinations $\mathbb{D} = \{\mathcal{D}_i, i = 1, 2, \ldots, N_{\mathcal{D}}\}$, it is necessary to introduce a mode variable, denoted as $l_k \in \mathbb{D}$, which specifies the underlying destination. Here, l_k is modelled by a semi-Markov process and therefore it remains unchanged within $(\tau_k, \tau_{k+1}]$. Consequently, the variable rate state vector at τ_k is given by

$$\mathbf{z}_k = [v_k, \psi_k, \mathbf{p}_k, l_k, T_{T,k}, T_{P,k}]',$$

from which kinematic states $(v, \psi, \mathbf{p})$ in $(\tau_k, \tau_{k+1}]$ can be obtained deterministically as per (3), (4) and (5). Now, we are in a position to give the state transition density associated with our system model:

$$p(\mathbf{z}_k, \tau_k | \mathbf{z}_{k-1}, \tau_{k-1}) = p(\mathbf{z}_k | \mathbf{z}_{k-1}, \tau_k, \tau_{k-1}) p(\tau_k | \tau_{k-1}). \tag{8}$$

Furthermore, the models in (7) need to be modified as

$$T_{T,k} | \mathbf{z}_{k-1}, l_k \sim \mathcal{N}(\mu_{T,k}, \sigma_T^2), T_{P,k} | \mathbf{z}_{k-1}, l_k \sim \mathcal{N}(\mu_{P,k}, \sigma_P^2). \tag{9}$$

It should be noted that continuity assumption has been imposed for the position (also for v, ψ) at τ_k, meaning that $\mathbf{p}_k = \mathbf{p}_{k^-} = \mathbf{p}_{k^+}$ where $\mathbf{p}_{k^-}$ and $\mathbf{p}_{k^+}$ are the positions immediately before and after τ_k. Note also that the condition on τ_k and τ_{k-1} has been made implicit. Moreover, there is no need to include $\hat{\mathbf{e}}_{T,k}$ and $\hat{\mathbf{e}}_{P,k}$ in the state vector since they can also be attained in a deterministic manner provided $\mathbf{z}_{k-1}$.

Finally, instead of using a fixed Markov chain transition matrix, the following transition density for l_k (note $l_k = l_{k^+}$) is applied

$$\begin{aligned} p(l_k | \mathbf{z}_{k-1}) &= p(l_k | l_{k-1}, \mathbf{p}_{k^-}, \psi_{k^-}) \\ &= \begin{cases} P_{UC}, & \text{if } l_k = l_{k-1} \\ (1 - P_{UC}) \hat{p}(l_k | \mathbf{p}_{k^-}, \psi_{k^-}), & \text{otherwise} \end{cases} \end{aligned} \tag{10}$$

where P_{UC} is the probability of leaving the intended destination unchanged while $\hat{p}(l_k | \mathbf{p}_{k^-}, \psi_{k^-})$ is a transition density in which information from heading angle and position is leveraged to guide changes in destination mode. We also note that a formulation similar to (10) has been used in [25] for modelling transition probabilities of group structure. More specifically, the density is given by

$$\hat{p}(l_k | \mathbf{p}_{k^-}, \psi_{k^-}) = \frac{\xi(l_k = \mathcal{D}_i | \mathbf{p}_{k^-}, \psi_{k^-})}{\sum_{\forall j, \mathcal{D}_j \neq l_{k-1}} \xi(l_k = \mathcal{D}_j | \mathbf{p}_{k^-}, \psi_{k^-})}$$

where

$$\begin{aligned} \xi(l_k = \mathcal{D}_i | \mathbf{p}_{k^-}, \psi_{k^-}) &= \exp\Big(-\frac{R_{OD}^2}{2\sigma_{OD}^2}\Big) \cdot \exp\Big(-\frac{R_{\psi}^2}{2\sigma_{\psi}^2}\Big), \\ R_{OD} &= \|\mathbf{p}_{\mathcal{D}_i} - \mathbf{p}_{k^-}\|, \\ R_{\psi} &= \arctan\Big(\frac{y_{\mathcal{D}_i} - y_{k^-}}{x_{\mathcal{D}_i} - x_{k^-}}\Big) - \psi_{k^-}, \end{aligned}$$

and $\{\sigma_{OD},\ \sigma_{\psi}\}$ are standard deviations that are fixed here. This density allows l_k to have a relatively high probability of being $\mathcal{D}_i$ when the object is facing and physically close to the destination.

2.3. Measurement Model

While the system model is designed in a particular way for the joint modelling of intent and kinematics, here we consider a general model for noisy measurements $\{\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_n\}$ of the hidden state made at $\{t_0 = 0, t_1, \ldots, t_n\}$:

$$\mathbf{y}_n \sim g(\mathbf{y}_n | \mathbf{z}_n) \tag{11}$$

where $g(\cdot)$ is some known probability density function and $\mathbf{z}_n$ is the state interpolated at t_n. For the convenient presentation of the deterministic mapping between $\mathbf{z}_n$ and $\mathbf{z}_k$, we further define

$$\mathbf{z}_n = \gamma_n(\mathbf{z}_k, t_n, \tau_k), \quad t_n \in (\tau_k, \tau_{k+1}].$$

Consequently, the conditional likelihood can be re-written as $g(\mathbf{y}_n | \gamma_n(\mathbf{z}_k, t_n, \tau_k))$. The overall system is therefore described by the system and measurement models in (8) and (11), respectively.

3. STATE AND INTENT ESTIMATION USING THE VARIABLE RATE PARTICLE FILTER

In this section we describe the inference method which allows for the on-line estimation of both the kinematic state and the intent of a moving object. To this end, we need to obtain the posterior distribution $\pi(\{\mathbf{z}_k, \tau_k\}_{k:t_0 \leq \tau_k < t_n} | \mathbf{y}_{0:n})$ recursively. Since it is impossible to produce a closed-form solution for this posterior distribution, we

resort to sampling-based methods. In particular, in this paper we implement the variable rate particle filter (VRPF) [18, 19, 26], which is a variant of particle filter tailored to variable rate models, to approximate the posterior distribution numerically.

The posterior distribution targeted by the VRPF at t_n can be factorised as

$$\begin{aligned}\pi(\{\mathbf{z}_k,\tau_k\}_{k:t_0\le\tau_k<t_n}|\mathbf{y}_{0:n}) &\propto g(\mathbf{y}_n|\gamma_n(\mathbf{z}_{K_n},t_n,\tau_{K_n}))\\ &\times p(\{\mathbf{z}_k,\tau_k\}_{k:t_{n-1}\le\tau_k<t_n}|\{\mathbf{z}_k,\tau_k\}_{k:t_0\le\tau_k<t_{n-1}})\\ &\times \pi(\{\mathbf{z}_k,\tau_k\}_{k:t_0\le\tau_k<t_{n-1}}|\mathbf{y}_{0:n-1}),\end{aligned}$$

with $K_n = \mathrm{Card}\{k, \forall\tau_k < t_n\}$. Sampling from the prior distribution $p(\{\mathbf{z}_k,\tau_k\}_{k:t_{n-1}\le\tau_k<t_n}|\{\mathbf{z}_k,\tau_k\}_{k:t_0\le\tau_k<t_{n-1}})$ can be accomplished by drawing $\{\mathbf{z},\tau\}$ from (8) sequentially until there is one pair whose arrival time falls after t_n and then keeping all drawn samples. Supposing that $\{\mathbf{z}_k^{(i)},\tau_k^{(i)}\}_{k:t_0\le\tau_k<t_{n-1}}$ are sampled from the following proposal distribution

$$\begin{aligned}&q(\{\mathbf{z}_k,\tau_k\}_{k:t_0\le\tau_k<t_{n-1}}|\mathbf{y}_{0:n}) \qquad (12)\\ &\approx\sum_{i=1}^{N_p} v_{n-1}^{(i)}\delta_{\{\mathbf{z}_k^{(i)},\tau_k^{(i)}\}_{k:t_0\le\tau_k<t_{n-1}}}(\{\mathbf{z}_k,\tau_k\}_{k:t_0\le\tau_k<t_{n-1}}),\end{aligned}$$

with N_p being the number of particles, $\{v_{n-1}^{(i)}\}_{1\le i\le N_p}$ the selection weights that can be chosen in a way to improve the filter performance (see, e.g., the auxiliary particle filter [27]) and $\sum_{i=1}^{N_p} v_{n-1}^{(i)} = 1$, the weight update equation for VRPF can be shown to be

$$\begin{aligned}\tilde{\omega}_n^{(i)} &= \frac{\pi(\{\mathbf{z}_k,\tau_k\}_{k:t_0\le\tau_k<t_n}|\mathbf{y}_{0:n})}{q(\{\mathbf{z}_k,\tau_k\}_{k:t_0\le\tau_k<t_n}|\mathbf{y}_{0:n})}\\ &\propto\frac{p(\{\mathbf{z}_k,\tau_k\}_{k:t_{n-1}\le\tau_k<t_n}|\{\mathbf{z}_k^{(i)},\tau_k^{(i)}\}_{k:t_0\le\tau_k<t_{n-1}})}{q(\{\mathbf{z}_k,\tau_k\}_{k:t_{n-1}\le\tau_k<t_n}|\{\mathbf{z}_k^{(i)},\tau_k^{(i)}\}_{k:t_0\le\tau_k<t_{n-1}},\mathbf{y}_{0:n})}\\ &\times\frac{\omega_{n-1}^{(i)}}{v_{n-1}^{(i)}}g(\mathbf{y}_n|\gamma_n(\mathbf{z}_{K_n},t_n,\tau_{K_n}))\end{aligned}$$

where $\tilde{\omega}_n^{(i)}$ is the unnormalised weight of the i^{th} particle at t_n and $\omega_{n-1}^{(i)}$ is the normalised particle weight at t_{n-1}. In this paper, we consider the use of a bootstrap filter and adopt the biased selection scheme as given in [18]. As a consequence, a simple weight update equation can be obtained:

$$\tilde{\omega}_n^{(i)} \propto \frac{\omega_{n-1}^{(i)}}{v_{n-1}^{(i)}}g(\mathbf{y}_n|\gamma_n(\mathbf{z}_{K_n},t_n,\tau_{K_n})). \qquad (13)$$

A VRPF algorithm given the new incoming measurement $\mathbf{y}_n$ at t_n is summarised in Algorithm 1. To alleviate particle degeneracy, we regenerate $\{\mathbf{z}_{K_{n-1}+1},\tau_{K_{n-1}+1}\}$ for each replicated particle from the selection step. This is achieved by sampling from a constrained prior $p(\mathbf{z}_k,\tau_k|\mathbf{z}_{k-1},\tau_{k-1},\tau_k>t_{n-1})$; see [18] for more details.

It is often of practical interest to estimate the quantities of interest at the discrete observation times ($\mathbf{z}_{0:n}$) rather than those at the variable rate state occurrence times ($\{\mathbf{z}_k\}_{k:t_0\le\tau_k<t_n}$). For continuous variables such as position we can compute their Minimum Mean Square Error (MMSE) estimates as follows

$$\hat{\mathbf{z}}_n^{\text{MMSE}} = \sum_{i=1}^{N_p}\omega_n^{(i)}\gamma_n(\mathbf{z}_{K_n}^{(i)},t_n,\tau_{K_n}^{(i)}),$$

Algorithm 1 Variable rate particle filter

for $i = 1 \to N_p$ **do**
  Draw $\{\mathbf{z}_k^{(i)},\tau_k^{(i)}\}_{k:t_0\le\tau_k<t_{n-1}}$ from (12); ▷ Selection
  Compute $u_{n-1}^{(i)} = \omega_{n-1}^{(i)}/v_{n-1}^{(i)}$;
  Sample ▷ Propagation
  $\{\mathbf{z}_k^{(i)},\tau_k^{(i)}\}_{k:t_{n-1}\le\tau_k<t_n} \sim p(\cdot|\{\mathbf{z}_k^{(i)},\tau_k^{(i)}\}_{k:t_0\le\tau_k<t_{n-1}})$;
  Extend particle trajectory $\{\mathbf{z}_k^{(i)},\tau_k^{(i)}\}_{k:t_0\le\tau_k<t_n}$;
  Compute $\tilde{\omega}_n^{(i)}$ according to (13);
end for
Normalise particle weights $\omega_n^{(i)} = \tilde{\omega}_n^{(i)}/\sum_{j=1}^{N_p}\tilde{\omega}_n^{(j)}$;

while for intent estimation we examine the probability of each possible destination $\mathcal{D}_j \in \mathbb{D}$ as per ($\mathbb{I}_{(\cdot)}$ is the indicator function),

$$p(l_n = \mathcal{D}_j|\mathbf{y}_{0:n}) \approx \sum_{i=1}^{N_p}\omega_n^{(i)}\mathbb{I}_{\mathcal{D}_j}(l_n^{(i)}).$$

4. RESULTS

A trajectory extracted from the UZH-FPV drone racing dataset [28] is used here to benchmark the proposed algorithm in a tracking scenario with an agile target. As shown in Figure 3, the trail consists of a few sharp turns, over short intervals, and the drone speed changes rapidly. This indicates that the target is undertaking fast manoeuvres and its underlying dynamics cannot be accurately described by a simple (nearly) constant velocity model. For intent recognition, we designate four artificial nominal destinations where $\mathcal{D}_3$ is the intended endpoint. Given the shape of the trajectory, it can be unreasonable to assume that the drone true intention can be revealed (even from a close visual inspection) before/until the last turn in the track (i.e. after nearly 80% of the journey time). Thus, for intent prediction (*not* state estimation), we only report aggregate results for the last 20% of the total track time (see Figure 3).

We evaluate the performance of the proposed approach for two measurement models: a) linear and Gaussian, and b) nonlinear. For comparison purposes, the following methods are also examined:

- OU-based predictor: the ERV model in [5].
- BD predictor: the pseudo-observation based BD method in [16] with a nearly constant acceleration (CA) model [22].
- A fixed 4-by-4 Markov chain transition matrix is used for l_k in the intent-driven intrinsic model with its diagonal elements (i.e. the probability with which l_k remains unchanged) being equal to 0.85 and off-diagonal elements being 0.05.

4.1. Linear and Gaussian Measurement Model

In the first experiment, observations are measurements of Cartesian position perturbed by Gaussian disturbances:

$$\mathbf{y}_n = \mathbf{p}_n + \mathbf{v}_n, \quad \mathbf{v}_n \sim \mathcal{N}(0,\sigma_v^2\mathbf{I}_{2\times2}),$$

with the standard deviation of noise $\sigma_v = 1.5$. The parameters used in the proposed intent-driven intrinsic model and VRPF algorithm are listed in Table 1. This encompasses the IDIM with a fixed transition matrix, except for parameters related to (10). For OU-ERV the standard deviation of the process noise σ_{ERV}, the restoring force parameter η_{ERV}, and the damping factor ρ_{ERV} are 5, 0.01

Table 1: Testing parameters for IDIM.

Symbol	λ, m	σ_T, σ_P, F	α, β	$P_{UC}, \sigma_{OD}, \sigma_\psi$	N_p
Value	$0.1, 100$	$200, 1500, 60$	$1, \frac{10}{3}$	$0.7, 35, 30$	1500

Table 2: Performance comparison across 50 Monte Carlo simulations, with the linear and Gaussian measurement model.

	IDIM	OU-ERV	BD-CA	IDIM w/ fixed trans. matrix
RMSE (m)	1.154	1.165	1.202	1.169
CPR (%)	67.2	79.2	79.1	27.1

and 0.2, respectively. As for BD-CA, the process noise standard deviation $\sigma_{\text{CA}} = 5$, the number of quadrature points $q = 15$ and the covariance modelling the orientation-extent of each endpoint is $\Sigma_i = 1.5^2\mathbf{I}_{2\times 2}$. Also, we set the arrival time prior in BD-CA to $p(\mathcal{T}|\mathcal{D}_i) = \text{Unif}(10\text{sec}, 30\text{sec})$. Furthermore, a uniform prior on destinations, i.e. $p(\mathcal{D} = \mathcal{D}_i) = 1/4$ for each $\mathcal{D}_i \in \mathbb{D}$, is applied in both OU-ERV and BD-CA. Given the linear Gaussian nature of the system, a Kalman filer-based inference routine is implemented here for the OU-ERV and BD-CA methods as in [5, 16].

The comparison below is for a set of $M = 50$ Monte Carlo simulations where the underlying (truth) trajectory remains identical while the observations are generated randomly according to the above measurement model. The metric used for the evaluation of tracking performance is the RMSE defined as follows:

$$\text{RMSE} = \sqrt{\frac{1}{MN}\sum_{m}^{M}\sum_{n}^{N}\|\hat{\mathbf{z}}_n^m - \mathbf{z}_n\|^2}.$$

Specifically, we examine the position RMSE for each algorithm. Note that in order to evaluate the tracking performance in OU-ERV, the posterior distribution for the hidden state is calculated according to $p(\mathbf{x}_n|\mathbf{y}_{0:n}) = \sum_{i=1}^{N_\mathcal{D}} p(\mathbf{x}_n|\mathbf{y}_{0:n}, \mathcal{D} = \mathcal{D}_i)p(\mathcal{D} = \mathcal{D}_i|\mathbf{y}_{0:n})$, i.e. Bayesian model averaging (BMA) [29]. For intent prediction, we compute the Maximum *a posteriori* (MAP) estimates,

$$\hat{I}_n = \text{argmax}_{\mathcal{D}_j\in\mathbb{D}}\, p(l_n = \mathcal{D}_j|\mathbf{y}_{0:n}),$$

in the last 20% of the time and compare the results against the true destination $\mathcal{D}_3$. The correct prediction rate (CPR) is then averaged across the M trails. The testing results are given in Table 2. It can be seen that our proposed method (IDIM) can deliver competitive tracking performance when compared with OU-ERV and BD-CA. We argue that this can be attributed to the fact that the intrinsic coordinate model provides a better approximation to the object motion in comparison to CA and ERV models. The position RMSE of OU-ERV is lower than that of the BD-CA as the model uncertainty is taken into account via BMA. In terms of intent prediction accuracy, IDIM is slightly outperformed by OU-ERV and BD-CA. This is expected as the latter two, which are based on Kalman filtering, are well-suited for linear Gaussian systems. Notably, IDIM with a fixed transition matrix for the mode variable cannot provide comparable destination prediction results, confirming that our proposal of using (10) is reasonable and can be highly beneficial. To visualise the IDIM performance over time, Figure 4 depicts the inference outcome for one of the realisations.

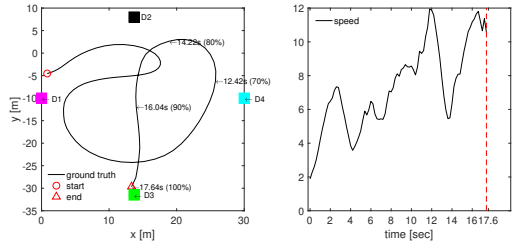

Fig. 3: Drone trajectory extracted from outdoor 45 degree downward facing [# 1] in [28]; $N = 89$ data points and $t_N = 17.6s$.

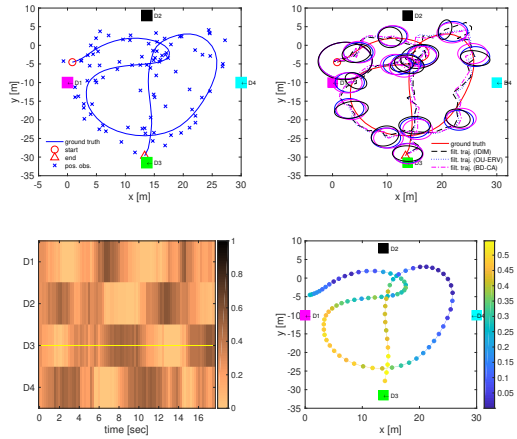

Fig. 4: Example result. Top left: true trajectory and observations; top right: posterior mean filtering estimates with confidence ellipses; bottom left: $p(l_n|\mathbf{y}_{0:n})$ from IDIM over time; bottom right; bottom left: $p(l_n = \mathcal{D}_3|\mathbf{y}_{0:n})$ from IDIM plotted along the path.

4.2. Nonlinear measurement model

Now, we consider the situation where observations are noisy range and bearing measurements from a radar sensor located at $[0,\ 0]'$:

$$\mathbf{y}_n = \begin{bmatrix}\|\mathbf{p}_n\| \\ \arctan(\frac{y_n}{x_n})\end{bmatrix} + \mathbf{v}_n, \quad \mathbf{v}_n \sim \mathcal{N}\big(0, \text{diag}([\sigma_r^2, \sigma_\theta^2])\big),$$

with $\sigma_r = 1.5$ and $\sigma_\theta = 5^\circ$. While IDIM requires no adaptation when used with nonlinear and/or non-Gaussian measurement models, OU-ERV and BD-CA need to be revised since they were introduced originally for linear and Gaussian systems. In order to assess the performance of OU-based and BD-based predictors, we replace Kalman filters with particle filters in both methods. Consequently, OU-ERV runs $N_\mathcal{D}$ particle filters in parallel to numerically evaluate $p(\mathbf{y}_{0:n}|\mathcal{D} = \mathcal{D}_i)$ whereas BD-CA utilises one particle filter to approximate $p(\mathbf{y}_{0:n}|\mathcal{D} = \mathcal{D}_i, \mathcal{T} = \mathcal{T}_j)$ for each destination $\mathcal{D}_i$ and quadrature point $\mathcal{T}_j$. In the experiment, 1500 particles are used per particle filter in OU-ERV and the resampling decision at a specific iteration will be made cooperatively by the filters (i.e. according to the global weights as in [30]). For BD-CA, we use 1500 particles for the single particle filter. Model parameters of IDIM, OU-ERV and BD-CA will remain unchanged as in the previous testing case.

Similarly, 50 realisations of the observation model are generated for performance evaluation, with the results summarised in Table 3. It can be noticed that without parameter tuning IDIM outperforms the other approaches in terms of both tracking and intent prediction

Table 3: Performance comparison across 50 Monte Carlo simulations, with the nonlinear measurement model.

	IDIM	OU-ERV	BD-CA
RMSE (m)	1.165	1.196	1.252
CPR (%)	61.9	25.6	59.4

accuracy under a more complex measurement model. Whilst prediction accuracy of both OU-ERV and BD-CA deteriorate, this is more prominent for OU-ERV with parallel particle filters. This highlights the need of OU-type intent predictor for more elaborate particle filtering schemes to support the likelihood estimation. Additionally, there is a trade-off between the computational cost and the performance of BD-CA since its complexity not only depends on the number of particles but also on $N_{\mathcal{D}}$ and q. IDIM with the fixed transition matrix is left out because its intent prediction performance is poor (even in the linear and Gaussian case).

5. CONCLUSION

In this paper, a new approach was introduced to capture the influence of the object intent on its motion using a physically well-justified dynamical model. An effective inference algorithm was subsequently developed to enable determining this underlying, *a priori* unknown and possibly time-varying, intent using variable rate particle filter. The presented numerical results on an agile target tracking example suggests that our method can be a competitive candidate for the simultaneous estimation of destination and kinematic state in challenging on-line tracking/surveillance scenarios.

6. REFERENCES

[1] Y. Bar-Shalom, P. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*, YBS Publishing, 2011.

[2] B. Völz, H. Mielenz, I. Gilitschenski, R. Siegwart, and J. Nieto, "Inferring pedestrian motions at urban crosswalks," *IEEE Transactions on Intelligent Transportation Systems*, 2018.

[3] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," *arXiv preprint arXiv:1905.06113*, 2019.

[4] S. Gaurav and B. Ziebart, "Discriminatively learning inverse optimal control models for predicting human intentions," in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, Richland, SC, 2019, AAMAS '19, pp. 1368–1376.

[5] B. I. Ahmad, J. K. Murphy, P. M. Langdon, and S. J. Godsill, "Bayesian intent prediction in object tracking using bridging distributions," *IEEE Transactions on Cybernetics*, vol. 48, no. 1, pp. 215–227, 2018.

[6] D. A. Castanon, B. C. Levy, and A. S. Willsky, "Algorithms for the incorporation of predictive information in surveillance theory," *International Journal of Systems Science*, vol. 16, no. 3, pp. 367–382, 1985.

[7] E. Baccarelli and R. Cusani, "Recursive filtering and smoothing for reciprocal Gaussian processes with Dirichlet boundary conditions," *IEEE Transactions on Signal Processing*, vol. 46, no. 3, pp. 790–795, 1998.

[8] L. M. Millefiori, P. Braca, K. Bryan, and P. Willett, "Modeling vessel kinematics using a stochastic mean-reverting process for long-term prediction," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 5, pp. 2313–2330, 2016.

[9] M. Üney, L. M. Millefiori, and P. Braca, "Data driven vessel trajectory forecasting using stochastic generative models," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8459–8463.

[10] M. Fanaswala and V. Krishnamurthy, "Spatiotemporal trajectory models for metalevel target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 30, no. 1, pp. 16–31, 2015.

[11] R. Rezaie and X. R. Li, "Dynamic models and representations of reciprocal and other Gaussian conditionally Markov sequences," *IEEE Transactions on Signal Processing*, 2019.

[12] V. Krishnamurthy and S. Gao, "Syntactic enhancement to VSIMM for roadmap based anomalous trajectory detection: A natural language processing approach," *IEEE Transactions on Signal Processing*, vol. 66, no. 20, pp. 5212–5227, Oct 2018.

[13] L. B. White and F. Carravetta, "State space realisations and optimal smoothing for gaussian generalised reciprocal processes," *IEEE Transactions on Automatic Control*, 2020.

[14] G. Zhou, K. Li, and T. Kirubarajan, "Constrained state estimation using noisy destination information," *Signal Processing*, vol. 166, pp. 107226, 2020.

[15] B. I. Ahmad, J. K. Murphy, P. M. Langdon, S. J. Godsill, R. Hardy, and L. Skrypchuk, "Intent inference for hand pointing gesture-based interactions in vehicles," *IEEE Transactions on Cybernetics*, vol. 46, no. 4, pp. 878–889, 2016.

[16] J. Liang, B. I. Ahmad, R. Gan, P. Langdon, R. Hardy, and S. Godsill, "On destination prediction based on Markov bridging distributions," *IEEE Signal Processing Letters*, vol. 26, no. 11, pp. 1663–1667, 2019.

[17] R. Gan, J. Liang, B. Ahmad, and S. Godsill, "Bayesian intent prediction for fast maneuvering objects using variable rate particle filters," in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2019.

[18] S. J. Godsill, J. Vermaak, W. Ng, and J. F. Li, "Models and algorithms for tracking of maneuvering objects using variable rate particle filters," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 925–952, 2007.

[19] P. Bunch and S. Godsill, "Dynamical models for tracking with the variable rate particle filter," in *2012 15th International Conference on Information Fusion*, 2012, pp. 1769–1775.

[20] S. McGinnity and G. W. Irwin, "Multiple model bootstrap filter for maneuvering target tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 3, pp. 1006–1012, 2000.

[21] M. S. Arulampalam, B. Ristic, N. Gordon, and T. Mansell, "Bearings-only tracking of manoeuvring targets using particle filters," *EURASIP Journal on Advances in Signal Processing*, vol. 2004, no. 15, 2004.

[22] X. R. Li and V. Jilkov, "Survey of maneuvering target tracking. Part I. Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.

[23] R. Best and J. Norton, "A new model and efficient tracker for a target with curvilinear motion," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 3, pp. 1030–1037, July 1997.

[24] M. Jacobsen, *Point process theory and applications: marked point and piecewise deterministic processes*, Boston: Birkhäuser, MA, 2006.

[25] S. K. Pang, J. Li, and S. J. Godsill, "Detection and tracking of coordinated groups," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 1, pp. 472–502, January 2011.

[26] M. R. Morelande and N. Gordon, "Rao-Blackwellised variable rate particle filters," in *2009 12th International Conference on Information Fusion*, 2009, pp. 1–8.

[27] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, 1999.

[28] J. Delmerico, T. Cieslewski, H. Rebecq, M. Faessler, and D. Scaramuzza, "Are we ready for autonomous drone racing? the UZH-FPV drone racing dataset," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6713–6719.

[29] J. A. Hoeting, D. Madigan, A. E. Raftery, and C. T. Volinsky, "Bayesian model averaging: A tutorial," *Statistical Science*, vol. 14, no. 4, pp. 382–417, 11 1999.

[30] L. Martino, J. Read, V. Elvira, and F. Louzada, "Cooperative parallel particle filters for online model selection and applications to urban mobility," *Digital Signal Processing*, vol. 60, pp. 172 – 185, 2017.