On Destination Prediction Based on Markov Bridging Distributions

Jiaming Liang, Bashar I. Ahmad, Runze Gan, Pat Langdon, Robert Hardy, and Simon Godsill

***Abstract*—This letter presents an alternative, more consistent, construction for bridging distributions, which enables inferring the destination of a tracked object from the available partial sensory observations. Two algorithms are then introduced to sequentially estimate the probability of all possible endpoints within a generic Bayesian framework. They capture the influence of intended destination on the object's motion via suitably adapted stochastic models. Whilst the bridging approach has low training requirements, the proposed formulation can lead to more efficient predictors, e.g. around 65% less computations for certain models. Synthetic and real data is used to illustrate the effectiveness of the introduced algorithms.**

***Index Terms*—intent inference, tracking, Kalman filter.**

I. Introduction

Knowing the destination of a tracked object can not only offer vital information on intent, enabling smart predictive functionalities and automation, but also facilitate more accurate state estimations, i.e. destination-aware tracking [1], [2]. It has various application areas such as smart navigation and trajectory planning for robots in the presence of other agents [3], [4], [5], intelligent interactive displays [6], revealing potential conflicts, patterns or anomalies in surveillance [7], [8], [9], driver assistance systems [10], [11], to name a few.

A. Problem Statement and Overall System Model

This letter addresses the problem of destination inference within a Bayesian framework and proposes two novel algorithms. It is emphasised that estimating the hidden state $x_t \in \mathbb{R}^s$, such as the object's position, velocity and higher order kinematics, is *not* sought here. Let $\mathbb{D} = \{\mathcal{D}_i : i = 1, 2, ..., N\}$ be the set of N nominal endpoints (e.g. harbours where a vessel can dock or selectable on-display icons) of a tracked object (e.g. vessel or pointing apparatus); each can be an extended region. The objective is to *sequentially* calculate the probability of each of $\mathcal{D}_i \in \mathbb{D}$ being the intended destination, thus $p(\mathcal{D} = \mathcal{D}_i \mid y_{1:k})$, $i = 1, ..., N$. The noisy sensory measurements at the time instant t_k are $y_{1:k} = \{y_1, y_2, ..., y_k\}$ pertaining to the consecutive instants $\{t_1, t_2, ..., t_k\}$.

A Gaussian Linear Time Invariant (LTI) formulation is adopted below since an approximate motion model that enables inferring the object's destination, rather than exact motion modelling, suffices. The state x_k at t_k is given by

$$x_k = F(h)x_{k-1} + M(h) + \varepsilon_k, \tag{1}$$

with $\varepsilon_k \sim \mathcal{N}(0, Q(h))$ a Gaussian dynamical noise. Matrices F and Q as well as vector M, which define the system model, are functions of the time step $h = t_k - t_{k-1}$.

Equation (1) encompasses any Gaussian LTI model, including those widely used in object tracking, e.g. the (near) Constant Velocity (CV), as well as mean reverting ones based on an Ornstein-Uhlenbeck (OU) process. The latter is described by: $dx_t = \Lambda(\mu_i - x_t)\, dt + \sigma dw_t$; its integration over interval $[t_{k-1}, t_k]$ produces (1) where vector M is a function of the process mean μ_i. Reversion strength is set by Λ and w_t is a Brownian motion. Observation $y_k \in \mathbb{R}^m$ is modelled by

$$y_k = Gx_k + \nu_k, \tag{2}$$

where G is a matrix mapping from the hidden state to the observed measurement and noise component $\nu_k \sim \mathcal{N}(0, V_k)$. A Gaussian distribution $\mathcal{D}_i \backsim \mathcal{N}(a_i, \Sigma_i)$ is assumed here to model an endpoint. The mean a_i and covariance Σ_i represent the centre and orientation-extent of $\mathcal{D}_i$, respectively. This is to maintain the linear Gaussian structure of the overall system.

B. Related Work and Contributions

Whilst the bridging distributions (BD) approach was introduced in [12], [13], a concise overview of its key results, including schemes for estimating future state and arrival time, was presented in [14]. BD in [12], [13], [14] captures the influence of endpoint $\mathcal{D}_i$ on the object motion by prescribing that the motion model in (1) has a terminal state x_K at arrival time $t_K = T$ equal to that of $\mathcal{D}_i$, i.e. $x_K \sim \mathcal{N}(a_i, \Sigma_i)$. Thereby, it constructs N bridged models. This implicitly assumes that x_K and the initial state x_1 at t_1 are independent. Although this can be approximately true in many scenarios, especially for $t_1 \ll T$, it is inconsistent with the Markov nature of (1) which dictates the transition density $p(x_K|x_1)$ and $p(x_K) = \int p(x_1)p(x_K|x_1)dx_1$.

In this paper, we introduce a different approach to [12], [13], [14] in which the destination point is considered to be a 'pseudo-observation' rather than a terminal state x_K of the system. Consequently, the mathematics of the dynamical model and observation process are made consistent with the Markov state process, in contrast with [12], [13], [14]. This new interpretation leads to two new destination prediction algorithms that can substantially reduce the computational complexity of the inference routine for all Gaussian LTI motion models in (1), e.g. by over 65% with Algorithm 1

This paragraph of the first footnote will contain the date on which you submitted your paper for review. This work was supported by Jaguar Land Rover via the CAPE Agreement.

J. Liang, B. I. Ahmad, R. Gan, P. Langdon, and S. Godsill are with the Engineering Department, University of Cambridge, Trumpington Street, Cambridge, CB2 1PZ. E-mails: {jl809, bia23, rg605, pml24, sjg30}@cam.ac.uk.

R. Hardy is with Jaguar Land Rover, Whitley, Coventry, UK. Email: rhardy@jaguarlandrover.com

in some cases (see Section II-C). Results from synthetic and real data demonstrate the effectiveness of these algorithms.

Several studies in the object tracking area consider incorporating, often known, destination information to improve the state estimation accuracy [1], [2]. Additionally, OU processes, with *a priori* learnt means, are shown to reliably model and estimate vessels motion in maritime surveillance [7], [8], [9]. Discretised state-space models based on reciprocal processes or other models from natural language processing are proposed in [15], [16] to recognise the object intent. They stipulate that the target should pass through a finite number of predefined spatial grid cells to reach its endpoint. The proposed approach in this letter aims to predict the unknown destination $\mathcal{D}$, not estimate x_n, and has notably lower complexity compared with those in [15], [16]. It utilises continuous-time state space models to treat asynchronous measurements and capture, via a Markov bridge, the long term underlying dependencies in the object trajectory as dictated by the intended endpoint.

Finally, various data driven prediction-classification methods rely on a dynamical model and/or *pattern of life* learnt from previously recorded data, e.g. [4], [5], [10], [11]. Whilst such techniques typically require substantial parameters training from extensive data sets (not always available), a probabilistic model-based framework is adopted here. It uses known dynamical and measurements models, with a few unknown parameters [17], [18]. Subsequently, an efficient inference approach, which requires minimal training, is introduced.

II. Bayesian destination Inference

Within a Bayesian framework, for each $\mathcal{D}_i \in \mathbb{D}$, we have

$$p(\mathcal{D}=\mathcal{D}_i|y_{1:k}) \propto p(\mathcal{D}=\mathcal{D}_i)p(y_{1:k}|\mathcal{D}=\mathcal{D}_i), \qquad (3)$$

where $p(\mathcal{D}=\mathcal{D}_i)$ is the prior on the i^{th} destination. This prior can be attained from relevant contextual information or learnt pattern-of-life. The objective of the inference module at t_k is hence to estimate the likelihoods $p(y_{1:k}|\mathcal{D}=\mathcal{D}_i)$, $i=1,2,...,N$. The key challenge here is to capture the influence of endpoint $\mathcal{D}_i$ on the object behaviour whilst utilising (1) and (2). Next, we describe a novel bridging formulation that facilitates introducing the conditioning on $\mathcal{D}_i$ in (3).

A. Pseudo-observation Formulation

The trajectory of the tracked object must end at the intended destination at arrival time T, albeit the exact path being random as per (1). A Markov bridge from t_k to T for $\mathcal{D}_i \in \mathbb{D}$ can be built to capture the influence of intent on the object motion by defining the pseudo-observation $\tilde{y}_K^i$ at $t_K = T$,

$$p(\tilde{y}_K^i = a_i|x_K, \mathcal{D}=\mathcal{D}_i) = \mathcal{N}(\tilde{y}_K^i = a_i \mid \tilde{G}x_K, \Sigma_i). \qquad (4)$$

Pseudo-observation matrix $\tilde{G}$ depends on the information available on the i^{th} endpoint $\mathcal{D}_i \backsim \mathcal{N}(a_i, \Sigma_i)$. For instance, $\tilde{y}_K^i$ may contain both position and velocity information.

For brevity of notation $\mathcal{D}=\mathcal{D}_i$ is replaced by $\mathcal{D}_i$ henceforth. Based on (4), we can express the *arrival-time-conditioned* likelihood at time instant t_k by

$$p(y_{1:k}|\mathcal{D}_i, T) = p(y_1|\mathcal{D}_i, T)\prod_{l=2}^{k} p(y_l|y_{1:l-1}, \mathcal{D}_i, T), \qquad (5)$$

such that $\tilde{y}_K^i$ introduces the conditioning on the endpoint $\mathcal{D}_i$ since $p(y_{1:k}|\mathcal{D}_i, T) = p(y_{1:k}|\tilde{y}_K^i = a_i, T)$. Whilst the arrival-time-conditioned likelihood $p(y_{1:k-1}|\mathcal{D}_i, T)$ pertaining to the previous time instant t_{k-1} is available at t_k, estimating the arrival-time-conditioned *Prediction Error Decompositions* (PEDs), i.e. $p(y_k \mid y_{1:k-1}, \mathcal{D}_i, T)$, suffices to sequentially calculate the likelihood in (5).

Since T is unknown, a prior distribution on T can be assumed in practice based on context, e.g. uniform where $p(T \mid \mathcal{D}_i) = \mathcal{U}(t_a, t_b)$ within the time window $\mathcal{T} = [t_a, t_b]$. The arrival time can then be marginalised out via

$$p(y_{1:k} \mid \mathcal{D}_i) = \int_{T\in\mathcal{T}} p(y_{1:k} \mid \mathcal{D}_i, T)p(T \mid \mathcal{D}_i)dT, \qquad (6)$$

to obtain the likelihood in (3). Since it is a one dimensional integral, a numerical approximation can be efficiently applied [19], e.g. Simpson's rule. This requires q evaluations of the arrival-time-conditioned PEDs $p(y_{1:k}|\mathcal{D}_i, T_n)$, for all $T_n \in \{T_1, T_2, ..., T_q\}$ which are drawn from the prior $p(T|\mathcal{D}_i)$.

Next, we we present two algorithms for estimating arrival-time-conditioned PEDs where the conditioning on T will be made implicity to simplify the notation. We show that they lead to a Kalman-filtering-type routine to predict $\mathcal{D}_I$.

B. Proposed Predictors (Algorithms 1 and 2)

The PED conditioned on the i^{th} endpoint and a given arrival time T can be written as

$$\begin{aligned} p(y_k|y_{1:k-1}, \mathcal{D}_i) &= \int p(y_k|x_k)p(x_k|y_{1:k-1}, \tilde{y}_K^i = a_i)dx_k, \\ &= \int p(y_k|x_k)\frac{p(x_k|y_{1:k-1})p(\tilde{y}_K^i = a_i|x_k)}{p(\tilde{y}_K^i = a_i|y_{1:k-1})}dx_k \qquad (7) \end{aligned}$$

which can be computed by parts and is dubbed Algorithm 1. Given the Gaussian linear nature of (1) and (2), the state posterior at the previous time instant t_{k-1} is given by: $p(x_{k-1}|y_{1:k-1}) = \mathcal{N}(x_{k-1}|\mu_{k-1|k-1}, \Sigma_{k-1|k-1})$ where *conveniently* $\mu_{k-1|k-1}$ and $\Sigma_{k-1|k-1}$ are the outputs of a Kalman Filter (KF), optimal estimate in the mean squared error sense [18]. Since $p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1}$, we have

$$p(x_k|y_{1:k-1}) = \mathcal{N}(x_k|\mu_{k|k-1}, \Sigma_{k|k-1}), \qquad (8)$$

with $\mu_{k|k-1} = F(h)\mu_{k-1|k-1} + M(h)$ and $\Sigma_{k|k-1} = F(h)\Sigma_{k-1|k-1}F(h)^T + Q(h)$. The pseudo-observation conditioned on the state at time instant t_k is

$$\begin{aligned} p(\tilde{y}_K^i = a_i|x_k) &= \int p(\tilde{y}_K^i = a_i|x_K)p(x_K|x_k)dx_K \\ &= \mathcal{N}(\tilde{y}_K^i = a_i|\mu_{\tilde{y}}, \Sigma_{\tilde{y}}) \qquad (9) \end{aligned}$$

where we have $\mu_{\tilde{y}} = \tilde{G}\left[F(T-t_k)x_k + M(T-t_k)\right]$ and $\Sigma_{\tilde{y}} = \tilde{G}Q(T-t_k)\tilde{G}^T + \Sigma_i$. This is based on transitional density $p(x_K|x_k) = \mathcal{N}\big(x_K|F(T-t_k)x_k + M(T-t_k), Q(T-t_k)\big)$ and also $p(\tilde{y}_K^i = a_i|x_K) = \mathcal{N}(\tilde{y}_K^i = a_i|\tilde{G}x_K, \Sigma_i)$. By utilising the following Gaussian identity

$$\mathcal{N}(x;\mu_1,\Sigma_1)\mathcal{N}(\mu_2;Lx,\Sigma_2) \propto \mathcal{N}(x;\mu_3,\Sigma_3),$$

with $\Sigma_3^{-1} = \Sigma_1^{-1} + L^T\Sigma_2^{-1}L,\ \mu_3 = \Sigma_3(\Sigma_1^{-1}\mu_1 + L^T\Sigma_2^{-1}\mu_2)$ and the *matrix inversion lemma* (Woodbury formula), the second component in the integral in (7) is given by

$$p(x_k|y_{1:k-1},\tilde{y}_K^i = a_i) \propto p(x_k|y_{1:k-1})p(\tilde{y}_K^i = a_i|x_k) = \mathcal{N}(x_k|\mu_*,\Sigma_*), \quad (10)$$

$$\mu_* = \mu_{k|k-1} + L_*\big[a_i - B_*\mu_{k|k-1} - \tilde{G}M(T-t_k)\big],$$
$$L_* = \Sigma_{k|k-1}B_*^T\big[B_*\Sigma_{k|k-1}B_*^T + \tilde{G}Q(T-t_k)\tilde{G}^T + \Sigma_i\big]^{-1},$$
$$\Sigma_* = \big(I - L_*B_*\big)\Sigma_{k|k-1}, \text{ and } B_* = \tilde{G}F(T-t_k).$$

From (7), the desired PED can then be obtained via

$$p(y_k|y_{1:k-1},\mathcal{D}_i) = \int \mathcal{N}(y_k|Gx_k,V_k) \times \mathcal{N}(x_k|\mu_*,\Sigma_*)dx_k = \mathcal{N}(y_k|\mu_y,\Sigma_y), \quad (11)$$

with $\mu_y = G\mu_*$ and $\Sigma_y = G\Sigma_*G^T + V_k$. At t_k the standard Kalman correction step is run to calculate the new posterior $p(x_k|y_{1:k})$ ready for the PED estimation at the next step t_{k+1}.

An alternative interpretation of endpoint inference using pseudo-observations can be achieved by re-writing the Bayesian smoothing (fixed-interval) equation, termed Rauch-Tung-Striebel smoother in linear Gaussian cases [20],

$$\begin{aligned} &p(x_{k-1}|y_{1:k-1},\tilde{y}_K^i = a_i) \\ &= \int \frac{p(x_{k-1}|y_{1:k-1})p(x_K|x_{k-1})}{p(x_K|y_{1:k-1})} p(x_K|y_{1:k-1},\tilde{y}_K^i = a_i)dx_K \\ &\propto p(x_{k-1}|y_{1:k-1}) \int p(\tilde{y}_K^i = a_i|x_K)p(x_K|x_{k-1})dx_K \\ &\propto \mathcal{N}(x_{k-1}|\mu_{k-1|k-1},\Sigma_{k-1|k-1})\mathcal{N}(\tilde{y}_K^i = a_i|\mu_{\tilde{y}},\Sigma_{\tilde{y}}) \\ &\quad = \mathcal{N}(x_{k-1}|\tilde{\mu},\tilde{\Sigma}), \end{aligned} \quad (12)$$

with $\{\mu_{\tilde{y}},\Sigma_{\tilde{y}}\}$ and $\{\tilde{\mu},\tilde{\Sigma}\}$ obtained similarly to those in (9) and (10). The PED can then be easily shown to reduce to

$$\begin{aligned} p(y_k|y_{1:k-1},\mathcal{D}_i) &= \int \Big[\int p(x_{k-1}|y_{1:k-1},\tilde{y}_K^i = a_i) \\ &\quad \times p(x_k|x_{k-1},\tilde{y}_K^i = a_i)dx_{k-1}\Big]p(y_k|x_k)dx_k \\ &= \mathcal{N}(y_k|\mu_y,\Sigma_y), \end{aligned} \quad (13)$$

$$\mu_y = G\Big[F(h)\tilde{\mu} + M(h) + L_y\Big[a_i - \tilde{G}\Big(F(T-t_k)\big(F(h)\tilde{\mu} - M(h)\big) + M(T-t_k)\Big)\Big]\Big],$$
$$\Sigma_y = G\big[A_y\tilde{\Sigma}A_y^T + (I - L_yB_y)Q(h)\big]G^T + V_k,$$
$$L_y = Q(h)B_y^T\big[B_yQ(h)B_y^T + \tilde{G}Q(T-t_k)\tilde{G}^T + \Sigma_i\big]^{-1},$$
$$A_y = \big(I - L_yB_y\big)F(h) \text{ and } B_y = \tilde{G}F(T-t_k).$$

The pseudo-code for Algorithms 1 and 2, which use (10)-(11) and (12)-(13), respectively, is shown below. The arrival-time-conditioned likelihoods are computed recursively as in (5) and T marginalisation in (6) is numerically approximated.

Algorithm 1 and 2 Inference with Pseudo-observations

Input: Obs.: $\{y_{1:k}\}$; Psedu-obs.: $\{\tilde{y}_{i,K},\Sigma_i\}_{1\leq i\leq N}$;
Initialisation: Mean $\mu_{1|1}$ and covariance $\Sigma_{1|1}$
for $k = 2 : K$ **do** ▷ For each timestamp
  KF predict and correct: $\mu_{k|k-1}$, $\Sigma_{k|k-1}$, $\mu_{k|k}$ and $\Sigma_{k|k}$
  for $i = 1 : N$ **do** ▷ For each destination
    for quadrature point $q = 1 : M$ **do**
      Compute: μ and Σ_* in (10); $\tilde{\mu}$ and $\tilde{\Sigma}$ in (12)
      Compute: $l_k^{i,q} = p(y_k|y_{1:k-1},\mathcal{D}_i)$ in (11); (13)
      Update T_q likelihood: $L_k^{i,q} = L_{k-1}^{i,q} \times l_k^{i,q}$
    end for
    Compute likelihood $P_k^i \approx p(y_{1:k}|\mathcal{D}_i)$ numerically.
  end for
  Obtain at t_k: $p(\mathcal{D} = \mathcal{D}_i|y_{1:k}) \approx \frac{P_k^i \times p(\mathcal{D}=\mathcal{D}_i)}{\sum_{j\in N} P_k^j \times p(\mathcal{D}=\mathcal{D}_i)}$
end for

C. Computational Complexity Analysis

The computational complexity is analysed here by counting each floating-point multiplication followed by one addition, i.e. a "flop" as in [21]. Let l be the dimensions of the pseudo-observation vector, whilst s and m are the dimensions of the state and measurement vectors, respectively. The computational cost of a Kalman filter at t_k is [22]: $\mathcal{C}_{\text{KF}}(m,s) = \frac{3}{2}(s^3 + s^2) + ms\big[\frac{3}{2}(s+m) + 3\big] + \frac{2}{3}(m^3 - m)$. Since the original BD in [13] runs one KF per endpoint with an extended state of dimensions $2 \times s$ and q points to approximate (6), its computational cost at a one time step assuming $l = s$ is

$$\mathcal{C}_{\text{BD}}(m,s) = qN\Gamma_{\text{BD}}, \quad (14)$$

where $\Gamma_{\text{BD}} = \frac{41}{2}s^3 + \frac{19}{2}s^2 - 2s + 3ms(2s+m+2) + \frac{2}{3}(m^3 - m)$. The complexities of the proposed Algorithms 1 and 2 are

$$\mathcal{C}_1(m,s) = qN\big[\Gamma_1(m,s) + \mathcal{C}_{\text{KF}}(m,s)\big], \quad (15)$$
$$\mathcal{C}_2(m,s) = qN\big[\Gamma_2(m,s) + \mathcal{C}_{\text{KF}}(m,s)\big], \quad (16)$$

respectively, such that

$$\Gamma_1(m,s) = \frac{43}{6}s^3 + \frac{9}{2}s^2 - \frac{2}{3}s + ms(s + \frac{1}{2}m + \frac{3}{2}),$$
$$\Gamma_2(m,s) = \frac{52}{3}s^3 + 12s^2 - \frac{4}{3}s + ms(s + \frac{1}{2}m + \frac{3}{2}).$$

This is for all models in (1), including those dependent on $\mathcal{D}_i$ such as OU-type models in [13]. For models independent of $\mathcal{D}_i$, e.g. CV, the introduced algorithms run one KF for all nominal destinations unlike the original BD. Their costs thereby reduce to: $\mathcal{C}_1(m,s) = qN\Gamma_1(m,s) + \mathcal{C}_{\text{KF}}(m,s)$ and $\mathcal{C}_2(m,s) = qN\Gamma_2(m,s) + \mathcal{C}_{\text{KF}}(m,s)$.

Table I depicts the complexity order of all methods as a function of the state dimension s; complexity order for m or any other parameter can be similarly attained from (15) and (16). As well as (14)-(16), the table clearly illustrates that the proposed formulation is significantly more efficient compared with the original BD. Algorithm 1 can reduce the computational complexity of the destination inference routine by approximately 57.5%; 65% for motion models independent of $\mathcal{D}_i$. As for $l < s$, i.e. only partial knowledge of the destination state is available, the computational complexity is reduced. In [23], a bridging-based method was proposed to reduce the inference complexity. It: a) uses the same formulation as in [13] which leads to inconsistencies, b) utilises crude approximations with unknown impact on the predictions quality, and c) only applies to models that are independent of

TABLE I: Complexity of BD methods as function of s; $c = 1.5s^3$.

Model (1)	Algorithm 1	Algorithm 2	Orig. BD
With $\mathcal{D}_i$	$\mathcal{O}(8.7qNs^3)$	$\mathcal{O}(18.8qNs^3)$	$\mathcal{O}(20.5qNs^3)$
No $\mathcal{D}_i$	$\mathcal{O}(7.2qNs^3 + c)$	$\mathcal{O}(17.3qNs^3 + c)$	$\mathcal{O}(20.5qNs^3)$

$\mathcal{D}_i$. Conversely, the methods presented here offer a consistent and efficient solution to the prediction problem without any approximations and it is applicable to any model in (1).

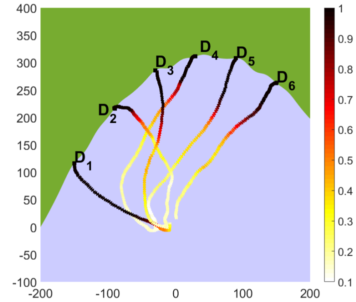
III. Numerical Examples

First, the destination probabilities $p(\mathcal{D} = \mathcal{D}_i|y_{1:k})$ estimated by the proposed algorithms is compared with that of the original BD in [13]. A maritime surveillance example is considered. The aim is to predict a vessel endpoint, out of $N = 6$ possible harbours in a bay, from noisy observations of its 2-D position, e.g. AIS-based. Synthetically generated trajectories from a CV model are utilised; all start from a rendezvousing area off the coast. We employ a CV model with dynamic noise parameter $\sigma = 0.1$, where σ^2 is the scaling factor of the standard CV covariance matrix [18], Simpson's quadrature scheme with $q = 15$ from a uniform prior $p(T|\mathcal{D}_i) = \mathcal{U}(80\text{mins}, 150\text{mins})$ and $p(\mathcal{D} = \mathcal{D}_i) = 1/6$ for $\mathcal{D}_i \in \mathbb{D}$. For each $\mathcal{D}_i \in \mathbb{D}$, a_i is the location of the centre of the i^{th} harbour and $\Sigma_i = \text{diag}[5I_2, 0.1I_2]$ describes its region.

Fig. 1 shows six synthetic tracks with the prediction results for the true destination. Whilst Fig. 1a demonstrates the effectiveness of the BD approach in predicting $\mathcal{D}$ (e.g. uncertainty diminishes as the target moves towards its endpoint and vice versa), Fig. 1b shows that the proposed algorithms give very similar results to that of the original BD. To confirm this, we generated 100 tracks and measured the Bhattacharyya distance $D_B = \frac{1}{8}(\mu_1 - \mu_2)^T\Sigma^{-1}(\mu_1 - \mu_2) + \frac{1}{2}\log\left(\frac{\det(\Sigma)}{\det(\Sigma_1)\det(\Sigma_2)}\right)$ between PEDs $p(y_k|y_{1:k-1}, \mathcal{D} = \mathcal{D}_i)$ and thereby $p(\mathcal{D} = \mathcal{D}_i|y_{1:k})$ for each $\mathcal{D}_i$. The means μ_m and covariances Σ_m, for $m = 1, 2$, pertain to the two compared Gaussian distributions and $\Sigma = (\Sigma_1 + \Sigma_2)/2$. The distances are averaged over the results of all $\mathcal{D}_i \in \mathbb{D}$ per track. It is displayed in Fig. 2, which illustrates that the difference between the outcomes of the proposed formulation and original BD is negligibly small; the results of the two introduced algorithms are nearly identical.

The run-times of MATLAB implementations of the three predictors are recorded for the above 100 tracks (System: *Intel(R) Core(TM) i7-4790 CPU@3.60GHz, 8GB RAM*). Algorithm 1 (mean run-time is $2.62ms$ at each t_k) shows a reduction of around 65% compared to the original BD (mean $7.42ms$); Algorithm 2 has mean run-time of $5.36ms$. This confirms the complexity analysis in (14)-(16) with parameters $\{m, s, q, N\} = \{2, 4, 15, 6\}$ and demonstrates the potential of the proposed efficient methods for real-time implementations.

Finally, we apply the three examined predictors to real data, namely 95 freehand pointing trajectories in 3-D. They were collected in an instrumented car with vision-based gesture tracker during driver/passenger interactions with the in-car touchscreen, at an average rate of $1/h = 50$ Hz. The objective is to predict, early in the pointing task, the intended on-display icon. A CV model with BD is used with uniform prior for all $N = 21$ selectable on-screen items. The performance across the three methods is assessed by the percentage of pointing time during which the true destination had the highest



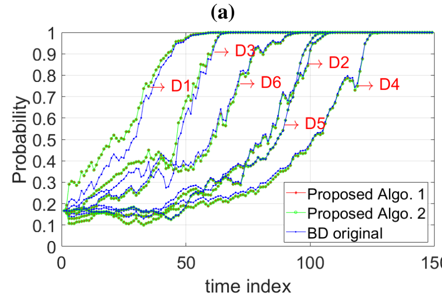


Fig. 1: Six synthetic vessel trajectories showing the calculated probability of the true destination. (a) Shows $y_{1:K}$ coloured by the prediction probability from Algorithm 1. (b) $p(\mathcal{D} = \mathcal{D}_i|y_{1:k})$ for original and proposed BD; D_X indicates the corresponding track.

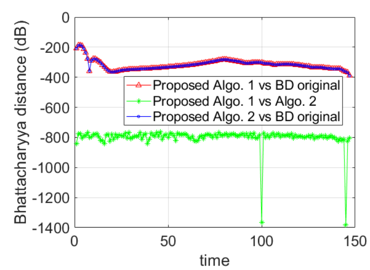


Fig. 2: Bhattacharyya distance in dB between $p(y_k|y_{1:k-1}, \mathcal{D} = \mathcal{D}_i)$ from different methods, averaged across all tracks and N endpoints.

estimated probability of being $\mathcal{D}_{\text{True}}$, i.e. with maximum *a posteriori* decision criterion from the estimated $p(\mathcal{D} = \mathcal{D}_i \mid y_{1:k})$, $i = 1, ..., N$. The results show that the two proposed algorithms and the original BD all have approximately 63% success rates, with $\sigma = 1$. This again demonstrates their similar performance. Reasonable changes to the CV dynamic noise, e.g. $\sigma \in [0.3\ \ 1.5]$, maintains a prediction success rates of $60 - 65\%$ for all BD approaches; values outside this range degrade the prediction accuracy since they represent more extreme deviations in pointing velocities. Whilst this confirms the BD insensitivity to reasonable changes to the model parameters as in [13], a detailed study of its robustness against model mismatches is outside the scope of this letter.

IV. Conclusions and Final Remarks

The proposed approach not only resolves the consistency issue with the previous bridging distributions construct, but also it can substantially reduce the computational complexity of predicting the target destination. Similar to BD in [12], [13], [14], it however requires prior knowledge of the location of all nominal endpoints, considers Gaussian linear set-ups and treats targets singly, even if they are a group. These can be addressed by extending the BD framework in future work.

REFERENCES

[1] 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.

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

[3] G. Best and R. Fitch, "Bayesian intention inference for trajectory prediction with an unknown goal destination," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015, pp. 5817–5823.

[4] H.-T. Chiang, N. Rackley, and L. Tapia, "Stochastic ensemble simulation motion planning in stochastic dynamic environments," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2015.

[5] K. Kitani, B. Ziebart, J. Bagnell, and M. Hebert, "Activity forecasting," in *Proc. of European Conf. on Computer Vision*, 2012, pp. 201–214.

[6] B. I. Ahmad, J. K. Murphy, S. Godsill, P. Langdon, and R. Hardy, "Intelligent interactive displays in vehicles with intent prediction: A Bayesian framework," *IEEE Signal Processing Magazine*, vol. 34, no. 2, pp. 82–94, 2017.

[7] 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.

[8] E. d'Afflisio, P. Braca, L. M. Millefiori, and P. Willett, "Maritime anomaly detection based on mean-reverting stochastic processes applied to a real-world scenario," in *Proc. of 21st International Conference on Information Fusion (FUSION)*, 2018, pp. 1171–1177.

[9] M. Uney, L. M. Millefiori, and P. Braca, "Prediction of rendezvous in maritime situational awareness," in *Proc. of 21st International Conference on Information Fusion (FUSION)*. IEEE, 2018, pp. 622–628.

[10] T. Bando, K. Takenaka, S. Nagasaka, and T. Taniguchi, "Unsupervised drive topic finding from driving behavioral data," in *Proc. of IEEE Intelligent Vehicles Symposium (IV)*, 2013, pp. 177–182.

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

[12] B. I. Ahmad, J. K. Murphy, P. M. Langdon, S. J. Godsill, and R. Hardy, "Destination inference using bridging distributions," in *Proc. of the 40th IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP '15)*, 2015, pp. 5585–5589.

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

[14] B. I. Ahmad, P. M. Langdon, and S. J. Godsill, "A Bayesian framework for intent prediction in object tracking," in *Proc. of the 44th IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'19)*, May 2019, pp. 8439–8443.

[15] 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.

[16] V. Krishnamurthy and S. Gao, "Syntactic enhancement to VSIMM for roadmap based anomalous trajectory detection: A natural language processing approach," *IEEE Trans. on Signal Processing*, 2018.

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

[18] A. J. Haug, *Bayesian Estimation and Tracking: A Practical Guide*. John Wiley & Sons, 2012.

[19] P. J. Davis and P. Rabinowitz, *Methods of numerical integration*. Courier Corporation, 2007.

[20] H. E. Rauch, C. Striebel, and F. Tung, "Maximum likelihood estimates of linear dynamic systems," *AIAA journal*, vol. 3, no. 8, pp. 1445–1450, 1965.

[21] O. Axelsson, *Iterative solution methods*. Cambridge university press, 1996.

[22] D. Willner, C.-B. Chang, and K.-P. Dunn, "Kalman filter configurations for multiple radar systems," Massachusetts Inst. of Tech. Lexington Lincoln Lab, Tech. Rep., 1976.

[23] T. Ardeshiri, B. I. Ahmad, P. M. Langdon, and S. J. Godsill, "Efficient bridging-based destination inference in object tracking," in *in Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP'17)*. IEEE, 2017, pp. 4406–4410.