Multiple Model Ballistic Missile Tracking with State-Dependent Transitions and Gaussian Particle Filtering

Miao Yu[@], Member, IEEE, Liyun Gong[@], Hyondong Oh[*], Member, IEEE, Wen-Hua Chen[+], Senior Member, IEEE, and Jonathon Chambers[#], Fellow, IEEE

Abstract

This paper proposes a new method for tracking the entire trajectory of a ballistic missile from launch to impact on the ground. Multiple state models are used to represent the different ballistic missile dynamics in three flight phases: boost, coast and reentry. In particular, the transition probabilities between state models are represented in a state-dependent way by utilising domain knowledge. Based on this modelling system and radar measurements, a state-dependent interacting multiple model approach based on Gaussian particle filtering is developed to accurately estimate information describing the ballistic missile such as the phase of flight, position, velocity and relevant missile parameters. Comprehensive numerical simulation studies show that the proposed method outperforms the traditional multiple model approaches for ballistic missile tracking.

Index Terms

Ballistic missile tracking, multiple state models, state-dependent transition probabilities, Bayesian inference, Gaussian particle filter

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/K014307/1 and the MOD University Defence Research Collaboration in Signal Processing

@ Miao Yu and Liyun Gong are with School of Computer Science, University of Lincoln, LN6 7TS, UK (e-mail: {myu, lgong}@lincoln.ac.uk).

∗ Hyondong Oh is with the School of Mechanical, Aerospace and Nuclear Engineering,Ulsan National Institute of Science and Technology (UNIST), 44919, Republic of Korea (e-mail: h.oh@unist.ac.kr).

+ Wen-Hua Chen is with Department of Aeronautical and Automotive Engineering, Loughborough University, LE11 3TU, UK (e-mail: w.chen@lboro.ac.uk).

Jonathon Chambers is with the School of Electrical and Electronic Engineering, Newcastle University, NE1 7RU, UK (e-mail: jonathon.chambers@ncl.ac.uk).

I. INTRODUCTION

A ballistic missile (BM) is one of the major threats from the air in modern warfare, so it is important to intercept before it hits the target on the ground. To intercept the BM, first it needs to be tracked by radar systems to estimate the state information such as position, velocity and other relevant parameters, based on which its future trajectory can be predicted by a corresponding dynamic model.

The BM typically experiences three different flight phases: boost, coast and reentry [1]. During those phases, the characteristics of the BM are significantly different: i) in the boost phase, the BM experiences a powered flight from launch to thrust cutoff; ii) in the coast phase, the thruster of the BM is turned off and the missile flies freely without the influence of atmospheric drag because it is in a relatively high part of the atmosphere; and iii) in the reentry phase, the BM reaches the lower part of the atmosphere and the atmospheric drag becomes considerable again and lasts until its impact to the ground.

Various works have considered BM tracking for the boost phase. A boost phase missile tracking algorithm is proposed in [2]. A nonlinear model is proposed to model the missile dynamics by correlating its transitional dynamics with the altitude motion and the line-of-sight angle is used as measurements. Based on the state model and measurements, the extended Kalman filter (EKF) is applied to estimate the state of a missile. An improved algorithm over [2] with better tracking performance is proposed in [3]. The batch based algorithm is used for the state initialisation and an adaptive process-noise matrix is added to compensate for the errors of the transition matrix in the dynamic model. A new dynamic model is proposed in [4], in which the thruster acceleration of the booster is modelled by a vector-differential equation that includes effects of both propellant depletion and attitude motions. The new model is incorporated into the EKF framework for the boost phase tracking. Li *et al.* [5] proposed a Maximum Likelihood (ML) algorithm for BM tracking at a particular acquisition time in the boost phase and the launch point. Based on the profile-based modelling of the boost phase and the line-of-sight measurements, the ML estimation method is applied for constructing and solving an optimisation function for estimating relevant parameters. A kind of adaptive filter algorithm is proposed in [6] for the boost-phase trajectory estimation. Polynomial model is used as the motion model of the boost trajectory and the corresponding process noise variance is constructed to make sure the state estimation error approximates the error lower bound of the optimal estimation. In order to achieve stably tracking the ballistic target and better adaptability to the flicker noise in the boost phase, a multiple model based method which combines the unscented Kalman filter and unscented particle filter as in [7] is proposed for tracking the ballistic missile in the boost phase.

There are also works for tracking the BM in the coast and reentry phase. Tracking of the BM in the coast phase is proposed in [8]. The sensor mechanism is modelled to deal with the time lag due to the mechanism of data collection and transmission and it is incorporated into the EKF for the state estimation. In the approach proposed in [9] for the coast phase tracking, the Doppler frequency is also taken into account for new measurement information. And different from the traditional Kalman filtering based approach, a unscented Klaman filtering (UKF) filtering approach is exploited for tracking. For the reentry phase tracking, an extended interval Kalman filter approach [10] and sequential Monte Carlo-based approach [11] have been developed by considering the effect of atmospheric drag. Besides, a comparison study between different filtering methods for BM tracking during the reentry phase is presented in [12]. From the numerical simulation results, it was shown that the Rao-Blackwellised particle filter achieves the best performance, especially when large initial uncertainties exist.

Note that the aforementioned methods only consider tracking of the BM during a particular phase by using one type of state models. However, in order to accurately track the whole trajectory of the BM, multiple state models need to be used as the BM experiences different flight phases from the launch to impact. To this end, Benavoli *et al.* [13] proposed an optimisation-based method to estimate the BM states and model parameters by adopting multiple models. A particle filtering-based approach has also been applied to estimate the burnout time. Different BM dynamic models (as detailed in [13]) have been designed to construct the cost function before and after the estimated burnout time and optimised for the state and parameter estimation. The limitation of this method is that it is always assumed that the tracking of a BM starts from the boost phase.

The most widely-used method for the tracking of multiple BM flight phases is the interacting multiple model (IMM) method as used in [14]–[17]. Multiple state models corresponding to different flight phases have been applied in the development of IMM algorithms where the state estimation is given by three steps: interaction, filtering and combination [18]. However, the current multiple model approaches still can not fully represent the real behaviour of a typical ballistic missile. For example, the IMM-based method uses a modelling system with *constant* transition probabilities between different models. This is not a realistic approach for BM tracking as the transitions between different phases are related with the states, that is, state-dependent. For instance, the higher the BM is, the more likely the BM flight phase transits from boost to coast.

In this paper, a new multiple model-based filtering approach is developed for BM tracking. Firstly, *a new state modelling framework with multiple models and state-dependent transition probabilities* is adopted for the BM movement modelling. The BM movement characteristics in different flight phases are reflected in

multiple models. Compared with the traditional multiple model-based BM tracking with constant transition probabilities, the state-dependent transition probabilities between different models are used in this work. Based on this modelling system, *state-dependent interacting multiple model Gaussian particle filtering (SD-IMMGPF)* approach is developed to implement the exact Bayesian inference framework. Different from the generic particle filtering-based state-dependent multiple model particle filtering (SD-IMMPF) ([19] and [20]), the proposed approach uses a modified version of the GPF [21] as mode-matched filtering. Compared with the SD-IMMPF, the proposed SD-IMMGPF approach can exploit both state model and measurement information for generating particles which can better approximate the posteriori state distribution for improving tracking results.

The structure of this paper is as follows. Section II describes the tracking models, including the proposed state modelling framework and the measurement model used in BM tracking. The general Bayesian inference procedure and the proposed SD-IMMGPF approach are presented in Section III. Comprehensive numerical simulation studies using different algorithms are presented in Section IV, and the final conclusions and suggestions for future work are given in Section V.

II. Ballistic missile tracking models

A. Multiple model system with state-dependent transition probabilities

This section presents the state modelling system used for the ballistic missile tracking. Multiple state models are applied for the different missile movements in different phases where the state transitions between them are represented in a state-dependent way.

1) Multiple state models: The entire trajectory of the BM from launch to impact is commonly divided into three phases [1], [13]: boost, coast and reentry phases as illustrated in Fig. 1. Thus, three state models are defined to reflect different BM dynamics. Similar to [13], we made the following assumptions: i) Earth is perfectly spherical and the rotation of the earth is considered; ii) the effect of the aerodynamic lift is currently neglected as in [10], [11], [13]; and iii) it is assumed that a single-stage boost phase with a constant thrust force exists.

Boost model

During the boost phase, the missile is affected by the gravity, thrust and aerodynamic drag force [13]. In an earth-centered-earth-fixed (ECEF) coordinate system [1], as the Earth rotates about the conventional terrestrial pole (CTP) axis with an angular velocity ω, the missile is also affected by two other forces:

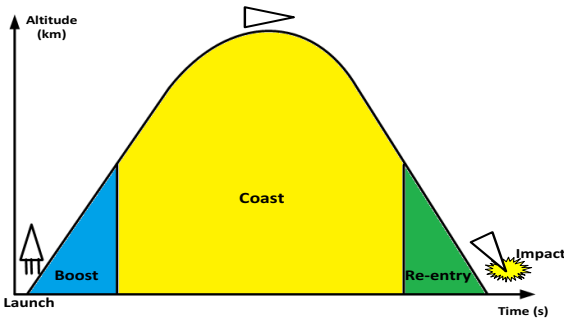


Fig. 1. The illustration of the entire trajectory and different phases of the BM.

coriolis and centripetal force. According to Newton's force law, the following basic equations hold:

$$\begin{aligned} \dot{\mathbf{p}}_t &= \mathbf{v}_t \\ \dot{\mathbf{v}}_t &= \mathbf{a}_t^{thrust} + \mathbf{a}_t^{drag} + \mathbf{a}_t^{gravity} + \mathbf{a}_t^{corriolis} + \mathbf{a}_t^{centripetal} \end{aligned} \tag{1}$$

where $\mathbf{p}_t = (p_t^x, p_t^y, p_t^z)^T$ and $\mathbf{v}_t = (v_t^x, v_t^y, v_t^z)^T$ ($(\cdot)^T$ denotes the vector transpose) represent the position and velocity in the ECEF coordinate system at the time t, respectively. Its z-axis is the CTP axis. The x and y axes lie in the equatorial plane with the x axis pointing towards the Greenwich meridian. The vectors $\mathbf{a}_t^{thrust}$, $\mathbf{a}_t^{drag}$, $\mathbf{a}_t^{gravity}$, $\mathbf{a}_t^{corriolis}$ and $\mathbf{a}_t^{centripetal}$ represent the accelerations introduced by thrust, aerodynamic drag, gravity, coriolis, and centripetal force, respectively.

As in [13] and [22], the thrust acceleration $\mathbf{a}_t^{thrust}$ acts along the target longitudinal axis (parallel to the velocity vector $\mathbf{v}_t$) and its magnitude is:

$$|\mathbf{a}_t^{thrust}| = \frac{gI_{sp}\dot{m}_t}{m_t} \tag{2}$$

where $m(t)$ is the target's mass, $g = 9.81ms^{-2}$ represent the gravitational acceleration, I_{sp} is the specific impulse (expressed in seconds) and $\dot{m}_t$ is the mass burn rate. Assuming that the specific impulse is constant and the target mass m_t decreases linearly at a constant rate $\dot{m}$ ($m_t = m_0 - \dot{m}_t$, m_0 is the target's mass at the launch time), the thrust acceleration magnitude can be expressed as

$$|\mathbf{a}_t^{thrust}| = \frac{ng}{1 - qt} \tag{3}$$

where $n = I_{sp}q$ is the initial thrust-to-weight ratio and $q = \frac{\dot{m}_t}{m_0}$ is the normalized mass burn rate.

The drag acceleration $\mathbf{a}_t^{drag}$ is opposite to the target's velocity vector $\mathbf{v}_t$. According to [22], its

magnitude is given by:

$$|\mathbf{a}_t^{drag}| = \frac{c_D(|\mathbf{v}_t|)S\rho(h_t)|\mathbf{v}_t|^2}{2m_t} \tag{4}$$

where $|\mathbf{v}_t|$ is the velocity magnitude at time t and h_t represents the altitude of the BM. S is defined as the target body cross-sectional area perpendicular to the velocity [22]. $c_D(|\mathbf{v}_t|)$ is the drag coefficient as a function of the velocity magnitude and $\rho(\cdot)$ is the air density function defined as:

$$\rho(h_t) = \rho_0 \exp(-k \cdot h_t) \tag{5}$$

where $\rho_0 = 1.22$ and $k = 0.14 \times 10^{-3}$.

By assuming $c_D(|\mathbf{v}_t|)S/m_t$ to be constant [13], a ballistic coefficient parameter $\beta = m_t/c_D(|\mathbf{v}_t|)S$ is introduced and (4) can be rewritten as:

$$|\mathbf{a}_t^{drag}| = \frac{\rho(h_t)|\mathbf{v}_t|^2}{2\beta} \tag{6}$$

The gravitational acceleration points from the target to the Earth's center and its magnitude is given by the Newton law of universal gravitation as ([13] and [22]):

$$|\mathbf{a}_t^{gravity}| = \frac{u_G}{|\mathbf{p}_t|^2} \tag{7}$$

where $u_G = 3.99 \times 10^{14} Nm^2/kg$ and $|\mathbf{p}_t|$ represents the position magnitude.

The coriolis and centripetal accelerations, $\mathbf{a}_t^{corriolis}$ and $\mathbf{a}_t^{centripetal}$ are defined as ([13] and [22]):

$$\begin{aligned} \mathbf{a}_t^{corriolis} &= 2\mathbf{w}_E \times \mathbf{v}_t \\ \mathbf{a}_t^{centripetal} &= \mathbf{w}_E \times (\mathbf{w}_E \times \mathbf{p}_t) \end{aligned} \tag{8}$$

where '$\times$' represents the cross product and $\mathbf{w}_E = (0, 0, -\omega)^T$. $\omega = 7.29 \times 10^{-5} rad/s$ is the Earth's angular speed.

Combining definitions of separate acceleration terms from (2) to (8), the total acceleration of the BM during the boost phase (denoted as $\mathbf{a}_t^b$) can be represented as:

$$\begin{aligned}\mathbf{a}_t^b &= \mathbf{a}_t^{thrust} + \mathbf{a}_t^{drag} + \mathbf{a}_t^{gravity} + \mathbf{a}_t^{corriolis} + \mathbf{a}_t^{centripetal} \\ &= \frac{ng}{1-qt}\frac{\mathbf{v}_t}{\|\mathbf{v}_t\|} - \frac{\rho(h_t)}{2\beta}\|\mathbf{v}_t\|\mathbf{v}_t - u_G\frac{\mathbf{p}_t}{\|\mathbf{p}_t\|} + 2\mathbf{w}_E \times \mathbf{v}_t + \mathbf{w}_E \times (\mathbf{w}_E \times \mathbf{p}_t)\end{aligned} \tag{9}$$

From the acceleration terms in (9) and the piecewise-constant acceleration assumption during a short time interval T, we can obtain the evolution of the position and velocity between t and $t+T$ as:

$$\begin{bmatrix} \mathbf{p}_{t+T} \\ \mathbf{v}_{t+T} \end{bmatrix} = F \begin{bmatrix} \mathbf{p}_t \\ \mathbf{v}_t \end{bmatrix} + G(\mathbf{a}_t^b + \mathbf{w}_t^b) \tag{10}$$

where $\mathbf{w}_t^b = (w_t^{x,b}, w_t^{y,b}, w_t^{z,b})^T$ represent the boost phase acceleration uncertainties in three axes and the matrix F and G are defined as:

$$F = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad G = \begin{bmatrix} \frac{T^2}{2} & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 \\ 0 & 0 & \frac{T^2}{2} \\ T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix} \tag{11}$$

Typically, as BM parameters n, q and β in (9) are unknown, they need to be estimated. The estimated parameters can then be used in missile trajectory prediction and missile type identification. In order to estimate the initial thrust-to-weight ratio n and normalised mass burn rate q, a simple Brownian motion model is used as:

$$\begin{aligned} n_{t+T} &= n_t + T \cdot w_t^n \\ q_{t+T} &= q_t + T \cdot w_t^q \end{aligned} \tag{12}$$

where n_t and q_t represent modeled n, q values at time instance t. w_t^n and w_t^q represent the introduced parameter uncertainties.

A similar way could be used to model the ballistic coefficient β. However, when the BM is at a high altitude, the value of $\frac{\rho(h_t)}{2\beta}$ in (9) will be close to zero regardless of β due to the exponential decay of the term $\rho(h_t)$ with respect to the height h_t. In this case, different values of β have the same effect on the position and velocity evolution, and thus the value of β can not be estimated correctly. In order to address this issue, we adopt the same strategy for parameter modelling used in [10]. Instead of β, a parameter $\gamma_t = \frac{\rho(h_t)}{2\beta}$ is first modelled and calculated. β can then be computed from γ_t. By the Euler

approximation [23], the evolution of γ_t can be modelled as:

$$\gamma_{t+T} = \gamma_t + T \cdot \gamma'_t + T \cdot w_t^{\gamma} \tag{13}$$

where w_t^{γ} represents the parameter uncertainty and γ'_t represents the differentiation of γ_t with respect to the time t given as:

$$\gamma'_t = -k \cdot \gamma_t \frac{p_t^x v_t^x + p_t^y v_t^y + p_t^z v_t^z}{\sqrt{(p_t^x)^2 + (p_t^y)^2 + (p_t^z)^2}}. \tag{14}$$

By augmenting the state dynamic equation (10) with the parameter models in (12) and (13), the complete state model for the boost phase is represented as:

$$\mathbf{x}_{t+T}^b = F^b \mathbf{x}_t^b + G^b \left(\begin{bmatrix} \mathbf{a}_t^b \\ \gamma'_t \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{w}_t^b \\ w_t^{\gamma} \\ w_t^n \\ w_t^q \end{bmatrix} \right) \tag{15}$$

where

$$\begin{aligned} \mathbf{x}_t^b &= \begin{bmatrix} p_t^x \\ p_t^y \\ p_t^z \\ v_t^x \\ v_t^y \\ v_t^z \\ \gamma_t \\ n_t \\ q_t \end{bmatrix}, \quad F^b = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\ G^b &= \begin{bmatrix} \frac{T^2}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{T^2}{2} & 0 & 0 & 0 \\ T & 0 & 0 & 0 & 0 & 0 \\ 0 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 0 & 0 & T \end{bmatrix} \end{aligned} \tag{16}$$

Coast and reentry models

After the boost phase, a BM will not be affected by the thrust force. The acceleration components in

three axes (denoted as $\mathbf{a}_t^{cr}$) become:

$$\mathbf{a}_t^{cr} = -\gamma_t \|\mathbf{v}_t\| \mathbf{v}_t - u_G \frac{\mathbf{p}_t}{\|\mathbf{p}_t\|} + 2\mathbf{w}_E \times \mathbf{v}_t + \mathbf{w}_E \times (\mathbf{w}_E \times \mathbf{p}_t). \tag{17}$$

When a BM is in the coast phase, it is at a high altitude and γ_t is a very small value. In this case, we model the γ_t to follow a Gaussian distribution with zero mean and very small standard deviation σ– $\gamma_t \sim N(0, \sigma^2)$. According to the definition of γ_t and the piecewise-constant acceleration assumption, the coast model is represented as:

$$\mathbf{x}_{t+T}^c = F^c \mathbf{x}_t^c + G^c \left(\begin{bmatrix} \mathbf{a}_t^{cr} \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{w}_t^c \\ w_t^{\gamma,c} \end{bmatrix} \right) \tag{18}$$

where

$$\mathbf{x}_t^r = \begin{bmatrix} p_t^x \\ p_t^y \\ p_t^z \\ v_t^x \\ v_t^y \\ v_t^z \\ \gamma_t \end{bmatrix}, \quad F^c = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad G^c = \begin{bmatrix} \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 & 0 \\ 0 & 0 & \frac{T^2}{2} & 0 \\ T & 0 & 0 & 0 \\ 0 & T & 0 & 0 \\ 0 & 0 & T & 0 \\ 0 & 0 & 0 & T \end{bmatrix} \tag{19}$$

where $\mathbf{w}_t^c$ is a 3×1 vector representing the coast model acceleration uncertainties and $w_t^{\gamma,c}$ is a scalar representing the uncertainty of γ_t in the coast model.

For the re-entry phase, the BM altitude decreases and the parameter γ_t is no longer negligible. Similar to the boost model, when the BM is within the lower part of the atmosphere, we model the evolution of γ_t in (13). The BM re-entry dynamic is then modelled as:

$$\mathbf{x}_{t+T}^r = F^r \mathbf{x}_t^r + G^r \left(\begin{bmatrix} \mathbf{a}_t^{cr} \\ \gamma_t' \end{bmatrix} + \begin{bmatrix} \mathbf{w}_t^r \\ w_t^{\gamma,r} \end{bmatrix} \right) \tag{20}$$

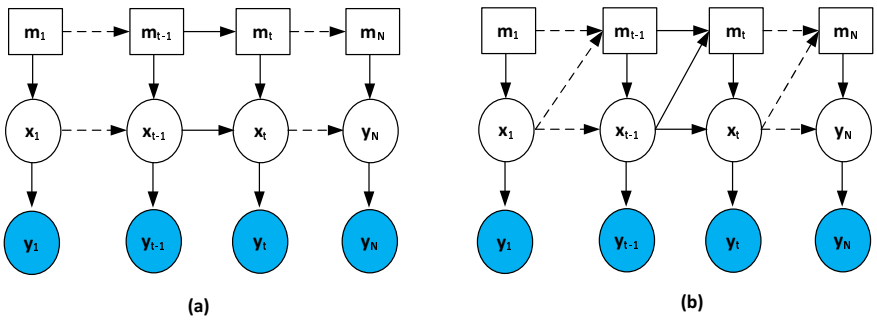


Fig. 2. The structure of a multiple model system with constant transition probabilities (a) and state dependent ones (b), with m_t, x_t and y_t representing the flight phase, state and measurement, respectively.

where

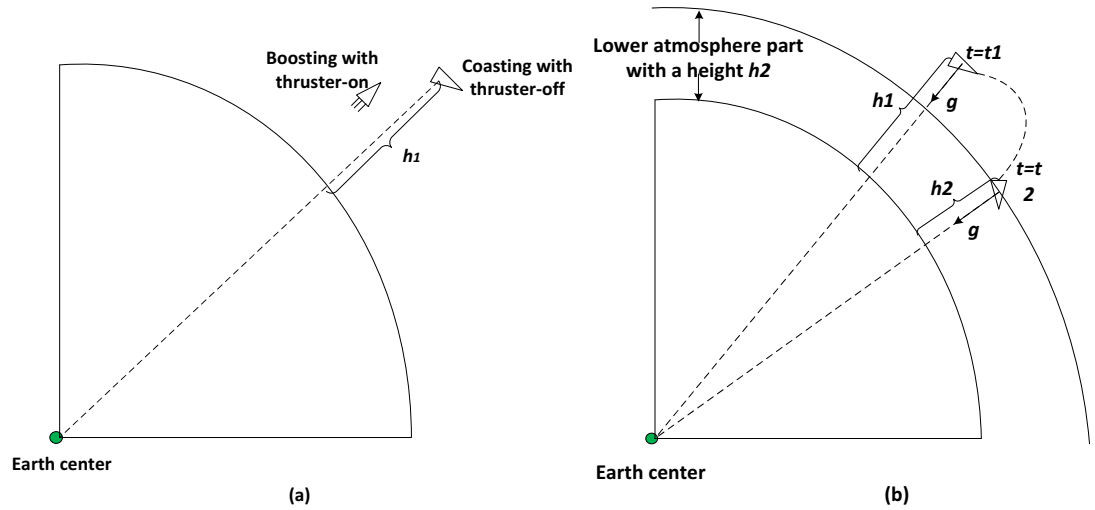
$$\mathbf{x}_t^r = \begin{bmatrix} p_t^x \\ p_t^y \\ p_t^z \\ v_t^x \\ v_t^y \\ v_t^z \\ \gamma_t \end{bmatrix}, \quad F^r = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad G^r = \begin{bmatrix} \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 & 0 \\ 0 & 0 & \frac{T^2}{2} & 0 \\ T & 0 & 0 & 0 \\ 0 & T & 0 & 0 \\ 0 & 0 & T & 0 \\ 0 & 0 & 0 & T \end{bmatrix} \tag{21}$$

and $\mathbf{w}_t^r$ is a 3×1 vector representing the re-entry model acceleration uncertainties and $w_t^{\gamma,r}$ is a scalar representing the uncertainty of γ_t in the reentry model.

2) State-dependent model transition probabilities: Transition probabilities between different flight phases (or the corresponding state models equivalently) can be represented as constant values in [14]–[17] where the current flight phase depends only on the one at the previous time instance, as illustrated in Fig. 2(a). However, in reality, the flight phase is also related to the state as represented in Fig. 2(b). Thus, the transition probabilities between different flight phases (or state models) are state dependent.

It is worthwhile noting that the transition between flight phases is dependent on the altitude information, as suggested in [1]. When the height reaches a particular threshold, the thruster of the BM is turned off and the flight phase transits to the coast phase, as illustrated in Fig. 3(a). As the missile flies in the coast phase, it first reaches a peak and then drops towards the ground due to the effect of gravity. When the altitude drops to a certain value, the BM reenters the low part of the atmosphere and transits to the reentry phase, as illustrated in Fig. 3(b).

This domain knowledge related to the flight phase transition and the altitude can be used to reflect the corresponding state model transitions as follows:



(a) The BM transits from boost to coast when it reaches a certain threshold with thruster being off

(b) As the BM approaches the lower part of the atmosphere from time t_1 to t_2, the height h reduce due to the effect of gravity.

Fig. 3. The transition of the BM between different phases.

$$\begin{aligned} m_t &= coast, \quad \text{if} \quad h_t > h_1 \quad \text{and} \quad m_{t-1} = boost \\ m_t &= reentry, \quad \text{if} \quad h_t < h_2 \quad \text{and} \quad m_{t-1} = coast \end{aligned} \tag{22}$$

where m_t represents the index of the state model ($boost$, $coast$ or $reentry$) related to the flight phase. The parameters h_t represents the ballistic missile height; h_1 and h_2 represent threshold values. Normally, the exact values of h_1 and h_2 are unknown, but some information could be obtained from previously collected information (e.g. the trajectory data collected for a particular missile type). The more information we obtain, the more accurate values can be obtained with less uncertainties.

In this work, to consider the uncertainties of h_1 and h_2, the Gaussian distribution is exploited to model them:

$$\begin{aligned} h_1 &\sim N(\cdot|m_{h_1}, \sigma^2_{h_1}) \\ h_2 &\sim N(\cdot|m_{h_2}, \sigma^2_{h_2}) \end{aligned} \tag{23}$$

where m_{h_1} and m_{h_2} represent the guess of the true values of h_1 and h_2, whilst σ_{h_1} and σ_{h_2} represent the standard deviations which represent the uncertainties for the height thresholds.

From (22) and (23), the transition probabilities from the boost to coast and from the coast to reentry

are modelled as (24), where $CDF(\cdot|m,\sigma^2)$ represents the cumulative density function for a Gaussian distribution with the mean m and standard deviation σ. In this way, the transition probabilities between different state models are modelled in a state-dependent way with respect to the h_t.

$$\begin{aligned} &p(m_t = coast|m_{t-1} = boost) = p(h_t > h_1) = CDF(h_t|m_{h1}, \sigma_{h1}^2) \\ &p(m_t = reentry|m_{t-1} = coast) = p(h_t < h_2) = 1 - CDF(h_t|m_{h2}, \sigma_{h2}^2) \end{aligned} \tag{24}$$

B. Measurement model

It is assumed that a radar measures the range r_t^m, azimuth angle θ_t^m and elevation angle φ_t^m of a BM in a local east-north-up (ENU) coordinate system [13]. The ENU coordinate system has the origin at the radar position, with three axes being towards the east, north and up directions, respectively. The global ECEF and local ENU coordinate systems are illustrated in Fig. 4, and the corresponding coordinates can be converted through:

$$\begin{bmatrix} p_t^e \\ p_t^n \\ p_t^u \end{bmatrix} = M \cdot \left(\begin{bmatrix} p_t^x \\ p_t^y \\ p_t^z \end{bmatrix} - \mathbf{p}^R \right) \tag{25}$$

where $[p_t^e, p_t^n, p_t^u]^T$ represents a position in the local ENU coordinate of the radar, $\mathbf{p}^R = \left[p^{x,R}, p^{y,R}, p^{z,R}\right]^T$ is the position of the radar in the ECEF coordinate system and M denotes the rotation matrix:

$$M = \begin{bmatrix} -sin(\lambda) & cos(\lambda) & 0 \\ -cos(\lambda)sin(\phi) & -sin(\lambda)sin(\phi) & cos(\phi) \\ cos(\lambda)cos(\phi) & sin(\lambda)cos(\phi) & sin(\phi) \end{bmatrix} \tag{26}$$

with ϕ and λ being the latitude and longitude of the radar. Under the local ENU coordinate system, the measurement equation is described as:

$$\begin{bmatrix} r_t^m \\ \theta_t^m \\ \varphi_t^m \end{bmatrix} = h(\mathbf{x}_t^s) + \mathbf{n}_t^m = \begin{bmatrix} \sqrt{(p_t^e)^2 + (p_t^n)^2 + (p_t^u)^2} \\ \arctan(\frac{p_t^n}{p_t^e}) \\ \arctan(\frac{p_t^u}{\sqrt{(p_t^e)^2 + (p_t^n)^2}}) \end{bmatrix} + \mathbf{n}_t^m \tag{27}$$

where $\mathbf{x}_t^s$ represents the state vector of a particular state model s,corresponding to the boost, coast or reentry phase as mentioned previously and $\mathbf{n}_t^m$ a measurement noise vector.

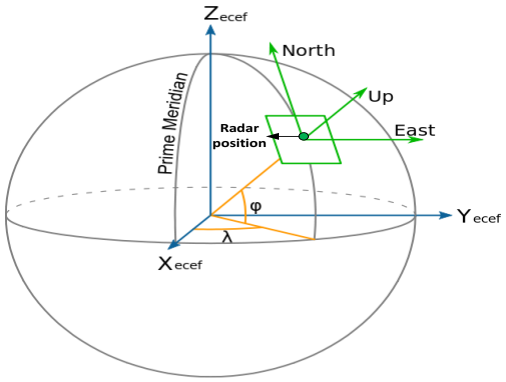


Fig. 4. The illustration of the global ECEF and local ENU coordinate systems.

III. State-dependent interacting multiple model Gaussian particle filtering

Based on different models defined in the previous section, a state-dependent interacting multiple model Gaussian particle filtering (SD-IMMGPF) algorithm is developed for ballistic missile tracking. It is based on the exact Bayesian inference framework for a multiple model system but with state-dependent transition probabilities.

A. Exact Bayesian framework for the multiple model system

The overall process of the exact Bayesian framework is divided into four steps:

$$p(m_{t-1}|\mathbf{Z}_{t-1}) \xrightarrow{\text{Mixing}} p(m_t|\mathbf{Z}_{t-1}) \tag{28}$$

$$p(\mathbf{x}_{t-1}|m_{t-1},\mathbf{Z}_{t-1}) \xrightarrow{\text{interacting}} p(\mathbf{x}_{t-1}|m_t,\mathbf{Z}_{t-1}) \tag{29}$$

$$p(\mathbf{x}_{t-1}|m_t,\mathbf{Z}_{t-1}) \xrightarrow{\text{Evolutions}} p(\mathbf{x}_t|m_t,\mathbf{Z}_{t-1}) \tag{30}$$

$$p(\mathbf{x}_t|m_t,\mathbf{Z}_{t-1}) \xrightarrow{\text{Correction}} p(\mathbf{x}_t,m_t|\mathbf{Z}_t) \tag{31}$$

where m_t denotes the model index, $\mathbf{x}_t$ the state vector and $\mathbf{Z}_t$ an ensemble of measurement vectors with $\mathbf{Z}_t = \{\mathbf{z}_1, ..., \mathbf{z}_t\}$.

1) Detailed Bayesian inference procedure: The Bayesian inferences for the four steps are described as follows.

Mode mixing: The mode mixing is related to the evolution of the model probability between consec-

utive discrete time instances $t-1$ and t. Using the law of total probability, we have:

$$p(m_t = s|\mathbf{Z}_{t-1}) = \sum_{r\in\mathcal{M}} p(m_t = s, m_{t-1} = r|\mathbf{Z}_{t-1}) = \sum_{r\in\mathcal{M}} p(m_t = s|m_{t-1} = r, \mathbf{Z}_{t-1})p(m_{t-1} = r|\mathbf{Z}_{t-1}),$$
$$\forall s, r \in \mathcal{M} = \{boost, coast, reentry\} \tag{32}$$

where $p(m_t = s|m_{t-1} = r, \mathbf{Z}_{t-1})$ can be further decomposed as:

$$p(m_t = s|m_{t-1} = r, \mathbf{Z}_{t-1}) = \int \pi_{rs}(\mathbf{x}_{t-1}) \cdot p(\mathbf{x}_{t-1}|m_{t-1} = r, \mathbf{Z}_{t-1})\, \mathrm{d}\mathbf{x}_{t-1} \tag{33}$$

where $\pi_{rs}(\mathbf{x}_{t-1})$ represents the state-dependent model transition probability from r to s.

State interacting: State interacting generates the initial mode-conditioned density $p(\mathbf{x}_{t-1}|m_t = s, \mathbf{Z}_{t-1})$ according to the conditional probability relation and the law of total probability as:

$$p(\mathbf{x}_{t-1}|m_t = s, \mathbf{Z}_{t-1}) = \frac{\sum_{r\in\mathcal{M}} \pi_{rs}(\mathbf{x}_{t-1}) \cdot p(\mathbf{x}_{t-1}, m_{t-1} = r|\mathbf{Z}_{t-1})}{p(m_t = s|\mathbf{Z}_{t-1})}. \tag{34}$$

Evolution: The state evolution step is to propagate the mode-conditioned state density from $t-1$ to t. Given the initial density provided in (34), the mode-conditioned prior distribution $p(\mathbf{x}_t|m_t = s, \mathbf{Z}_{t-1})$ at t is calculated as:

$$p(\mathbf{x}_t|m_t = s, \mathbf{Z}_{t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1}, m_t = s, \mathbf{Z}_{t-1})p(\mathbf{x}_{t-1}|m_t = s, \mathbf{Z}_{t-1})\, \mathrm{d}\mathbf{x}_{t-1} \tag{35}$$

where $p(\mathbf{x}_t|\mathbf{x}_{t-1}, m_t = s, \mathbf{Z}_{t-1})$ depends on the state model $m_t = s$.

Correction: Finally, the updated measurement is incorporated to correct the prior by Bayes rule:

$$p(\mathbf{x}_t, m_t = s|\mathbf{Z}_t) \propto p(\mathbf{z}_t|\mathbf{x}_t, m_t = s)p(\mathbf{x}_t|m_t = s, \mathbf{Z}_{t-1}) \cdot p(m_t = s|\mathbf{Z}_{t-1}). \tag{36}$$

The state estimation can then be derived from the updated posterior distribution $p(\mathbf{x}_t, m_t = s|\mathbf{Z}_t)$.

B. SD-IMMGPF implementation

There is no analytical solution for the exact Bayesian inference framework due to the nonlinearity and non-Gaussian distribution of the multiple model system. Thus, a particular implementation method is needed to obtain the approximated solution of the posterior state distribution in (36). Considering the state-dependent transition probabilities in the Bayesian inference framework, the conventional IMM filtering method in [14]–[17] is not suitable since it assumes the constant transition probabilities. In [19] and [20],

a particle filtering-based SD-IMMPF approach is proposed in order to implement the aforementioned Bayesian inference framework. However, in the SD-IMMPF approach, only the state model is applied for new particle generation; thus it is likely to obtain outliers (more details are explained below).

Instead of the SD-IMMPF, this study proposed a Gaussian particle filtering-based SD-IMMGPF for implementing the Bayesian inference to increase the sampling efficiency and tracking performance. Compared with the SD-IMMPF, the SD-IMMGPF applies Gaussian particle filtering-based approach for every mode-matched filter to generate particles better approximating the state posterior distribution (36). The details of the SD-IMMGPF approach are shown as follows.

Initially, it starts at time $t-1$ with the set of weighted particles $\{\mathbf{x}_{t-1}^{r,k}, w_{t-1}^{r,k}; r \in \mathcal{M}, k \in \{1,\ldots,N\}\}$ to approximate the probability $p(\mathbf{x}_{t-1}, m_{t-1} = r|\mathbf{Z}_{t-1})$.

Mode mixing implementation: Prior mode probability in (32) is approximated with generated particles as:

$$p(m_t = s|\mathbf{Z}_{t-1}) \approx \sum_{r\in\mathcal{M}} \sum_{k=1}^{N} \pi_{rs}(\mathbf{x}_{k-1}^{r,k}) \cdot w_{t-1}^{r,k} \triangleq \Lambda_{t-1}^{s}, \tag{37}$$

where Λ_{t-1}^{s} is defined to facilitate the rest of the derivation.

State interacting implementation: The state interacting process can be implemented by inserting particles at $t-1$ with the different mode index r, into (34) such that

$$p(\mathbf{x}_{t-1}|m_t = s, \mathbf{Z}_{t-1}) \approx \sum_{r\in\mathcal{M}} \sum_{k=1}^{N} \pi_{rs}(\mathbf{x}_{t-1}^{r,k}) w_{t-1}^{r,k} \delta(\mathbf{x}_{t-1} - \mathbf{x}_{t-1}^{r,k}) / \Lambda_{t-1}^{s}. \tag{38}$$

Evolution and correction implementation: In the SD-IMMPF method proposed in [19], a generic particle filtering-based approach is applied as the mode-matched filter to obtain the approximation of the posterior distribution. Firstly, the resampling method is applied to obtain a set of N particles $\{\mathbf{x}_{t-1}^{s,k}\}_{k=1,\ldots,N}$ from (38), based on which new particles $\{\mathbf{x}_{t}^{s,k}\}_{k=1,\ldots,N}$ are then predicted according to the state model corresponding to mode s. Weights of particles $\{w_{t}^{s,k}\}_{k=1,\ldots,N}$ are calculated by the likelihood function. The posterior distribution of (36) is then approximated by the obtained $\{w_{t}^{s,k}, \mathbf{x}_{t}^{s,k}\}_{k=1,\ldots,N}$ for every mode s value. However, the limitation of the SD-IMMPF method is that particles are only generated from the state model and the generated particles are likely to be outlier with low likelihood probability (as mentioned in [24]), which deteriorates the tracking performance. When the initial condition is not accurate enough and the number of particles is small, the performance of the SD-IMMPF algorithm is rather poor (as will be shown in the simulation studies).

In order to address this limitation of the SD-IMMPF, the Gaussian particle filtering (GPF) [21] based approach is applied for mode-matched filtering. Conditioned on a particular mode, a new importance

function which is a Gaussian approximation of the mode-based posterior distribution is constructed, by exploiting information in both the state and measurement models. In this way, particles which have higher likelihood values can be sampled from the constructed importance function to better approximate the related posterior distribution of (36), leading to more accurate state estimation. Besides, compared with other variants of particle filtering which also exploits state and measurement models for sampling particles (such as unscented particle filtering (UPF) [25]), the GPF based implementation is time efficient. The reason is that rather than constructing important functions for every particle (as in UPF, for every particle an important function needs to be constructed by the unscented Kalman filtering for sampling), only one important function needs to be constructed for every mode for particles generation.

Firstly, the mean and covariance for a Gaussian distribution to approximate $p(\mathbf{x}_{t-1}|m_t = s, \mathbf{Z}_{t-1})$ can be obtained as:

$$\begin{aligned} \mu_{t-1}^{s} &= \sum_{r\in\mathcal{M}}\sum_{k=1}^{N} \pi_{rs} w_{t-1}^{r,k} \mathbf{x}_{t-1}^{r,k} \\ \Sigma_{t-1}^{s} &= \sum_{r\in\mathcal{M}}\sum_{k=1}^{N} \pi_{rs} w_{t-1}^{r,k} (\mathbf{x}_{t-1}^{r,k} - \mu_{t-1}^{s}) \cdot (\mathbf{x}_{t-1}^{r,k} - \mu_{t-1}^{s})^{T} \end{aligned} \tag{39}$$

Based on this mean and covariance, we obtain a Gaussian approximation of the distribution $p(\mathbf{x}_t|m_t = s, \mathbf{Z}_t)$. Different methods can be applied to obtain such an approximation; in our work, the extended Kalman filter (EKF) is applied considering its efficiency and successful applications in the posterior distribution approximation of BM tracking [14]–[17]. The EKF in the GPF consists of the two steps: prediction and update. The prediction step predicts the mean and covariance by a particular state model as:

$$\mu_{t|t-1}^{s} = f^{s}(\mu_{t-1}^{s}) \tag{40}$$

where $f^{s}(\cdot)$ represents the state transition function corresponding to a particular mode s, from (15), (17) and (20) for the BM tracking problem.

$$\Sigma_{t|t-1}^{s} = (J_t^s)|_{\mu_{t-1}^{s}} \Sigma_{t-1}^{s} ((J_t^s)|_{\mu_{t-1}^{s}})^{T} + Q_s \tag{41}$$

where $(J_t^s)|_{\mu_{t-1}^{s}}$ represents the $(s$-th$)$ model first order Jacobian matrix value of mode s, at the initial mean value μ_{t-1}^{s}. The Q_s matrix is the covariance of the noise vector for the mode s. The mean and

covariance are then updated from the predicted results:

$$\begin{aligned} S_t^s &= H_t|_{\mu_{t|t-1}^s} \Sigma_{t-1}^s (H_t|_{\mu_{t|t-1}^s})^T + R \\ W_t^s &= \Sigma_{t|t-1}^s (H_t|_{\mu_{t|t-1}^s})^T (S_t^s)^{-1} \quad \text{(Kalman gain)} \\ \mathbf{r}_t^s &= \mathbf{z}_t - h(\mu_{t|t-1}^s) \quad \text{(measurement residual)} \\ \mu_t^s &= \mu_{t|t-1}^s + W_t^s \mathbf{r}_t^s \quad \text{(mean)} \\ \Sigma_t^s &= \Sigma_{t|t-1}^s - W_t^s \Sigma_{t|t-1}^s (W_t^s)^T \quad \text{(covariance)} \end{aligned} \tag{42}$$

where $h(\cdot)$ is the measurement model function in (27). The matrix $H_t|_{\mu_{t|t-1}^s}$ represents the value of the first order Jacobian matrix related to the measurement model function at $\mu_{t|t-1}^s$. The matrix R represents the measurement noise covariance. A Gaussian distribution is then obtained with the mean μ_t^s and covariance Σ_t^s, which is applied to approximate the posterior $p(\mathbf{x}_t|m_t = s, \mathbf{Z}_t)$.

A new set of particles $\{\mathbf{x}_t^{i,s}\}_{i=1,\ldots,N}$ is then sampled from this Gaussian distribution represented as $N(\mathbf{x}_t^{i,s}|\mu_t^s, \Sigma_t^s)$, which is constructed considering both the state model and measurement model. In this way, measurement information is considered for the particle generation, and thus generated particles will be more likely in a high measurement likelihood region. From the concept of importance sampling in [24] and (36), the posterior distribution $p(\mathbf{x}_t, m_t = s|\mathbf{Z}_t)$ is approximated as:

$$p(\mathbf{x}_t, m_t = s|\mathbf{Z}_t) \approx \sum_i w_t^{i,s} \delta(\mathbf{x}_t - \mathbf{x}_t^{i,s}) \tag{43}$$

with particle weights $\{w_t^{i,s}\}_{i=1,\ldots,N}$ being estimated as:

$$w_t^{i,s} \propto \frac{p(\mathbf{z}_t|\mathbf{x}_t^{i,s}, m_t = s) N(\mathbf{x}_t^{i,s}|\mu_{t|t-1}^s, \Sigma_{t|t-1}^s) p(m_t = s|\mathbf{Z}_{t-1})}{N(\mathbf{x}_t^{i,s}|\mu_t^s, \Sigma_t^s)} \tag{44}$$

where $N(\mathbf{x}_t^{i,s}|\mu_{t|t-1}^s, \Sigma_{t|t-1}^s)$ is a Gaussian approximation of $p(\mathbf{x}_t|m_t = s, \mathbf{Z}_{t-1})$. From the obtained particles and corresponding weights, both the state estimation and model probability can be estimated. The procedure of the SD-IMMGPF algorithm is summarised in Algorithm I.

IV. Numerical simulation studies

In this section, numerical simulation studies are performed to analyse the performance of the proposed SD-IMMGPF method for the BM tracking in terms of estimating mode probabilities, BM states and parameters. An entire BM trajectory is simulated in the earth-centred-earth-fixed (ECEF) coordinate system in Fig. 5. Key parameters of the simulated BM flight trajectory are listed in Table I, which

Algorithm 1 Summary of the SD-IMMGPF algorithm

Initially, it starts at time $t-1$ with the set of weighted particles $\{\mathbf{x}_{t-1}^{r,k}, w_{t-1}^{r,k}; r \in \mathcal{M}, k \in \{1,\ldots,N\}\}$ to approximate the probability $p(\mathbf{x}_{t-1}, m_{t-1} = r|\mathbf{Z}_{t-1})$.

- Mode mixing implementation:
  The prior mode probability $p(m_t = s|\mathbf{Z}_{t-1})$ is computed by (37).
- State interacting implementation:
  $p(\mathbf{x}_{t-1}|m_t = s, \mathbf{Z}_{t-1})$ is approximated by particles $\{\mathbf{x}_{t-1}^{r,k}, w_{t-1}^{r,k}; r \in \mathcal{M}, k \in \{1,\ldots,N\}\}$ using (38)
- Importance sampling function construction:
  (i) For every mode s, initial mean μ_{t-1}^s and covariance Σ_{t-1}^s for Gaussian approximation of $p(\mathbf{x}_{t-1}|m_t = s, \mathbf{Z}_{t-1})$ are estimated by (39)
  (ii) The extended Kalman filtering procedure is performed according to prediction ((40), (41)) and update (42) to obtain a Gaussian approximation of $p(\mathbf{x}_t|m_t = s, \mathbf{Z}_t)$, with mean μ_t^s and covariance Σ_t^s.
- Particles sampling and weights calculation:
  N Particles are generated from the importance function by $\mathbf{x}_t^{i,s} \sim N(\mathbf{x}|\mu_t^s, \Sigma_t^s)$ for $i = 1, ..., N$ and related weights $\{w_t^{i,s}\}_{i=1}^N$ are computed by (44).

Finally, according to the particles and weights, the state is estimated as:

$$\hat{\mathbf{x}}_t = \sum_{s\in\mathcal{M}} \sum_{i=1}^{N} w_t^{i,s} \mathbf{x}_t^{i,s} \tag{45}$$

and the probability of a particular mode $m_t = s$ is calculated as:

$$p(m_t = s) = \sum_{i=1}^{N} w_t^{i,s} \tag{46}$$

corresponds to the short range ballistic missile as described in [26]. Based on the simulated BM trajectory, algorithms can be applied for the BM tracking, with the following settings.

TABLE I
THE PARAMETERS OF THE SIMULATED BM TRAJECTORY

Flight time	Range	Boost time	Engine-off velocity
305 (s)	292 (km)	66 (s)	1.46 (km/s)

Initialisation: Considering the uncertainty about the initial state vector, Gaussian distributions are applied to model different components of the initial state vector. The initial position $\mathbf{p}_0$ and velocity $\mathbf{v}_0$ in the ECEF coordinate system can be modelled as:

$$\mathbf{p}_0 \sim N(\cdot|\overline{\mathbf{p}}_0, \Sigma_0^{\mathbf{p}}), \quad \mathbf{v}_0 \sim N(\cdot|\overline{\mathbf{v}}_0, \Sigma_0^{\mathbf{v}}) \tag{47}$$

where the means $\overline{\mathbf{p}}_0$ and $\overline{\mathbf{v}}_0$ represent the initial guess of the true position and velocity, respectively. The

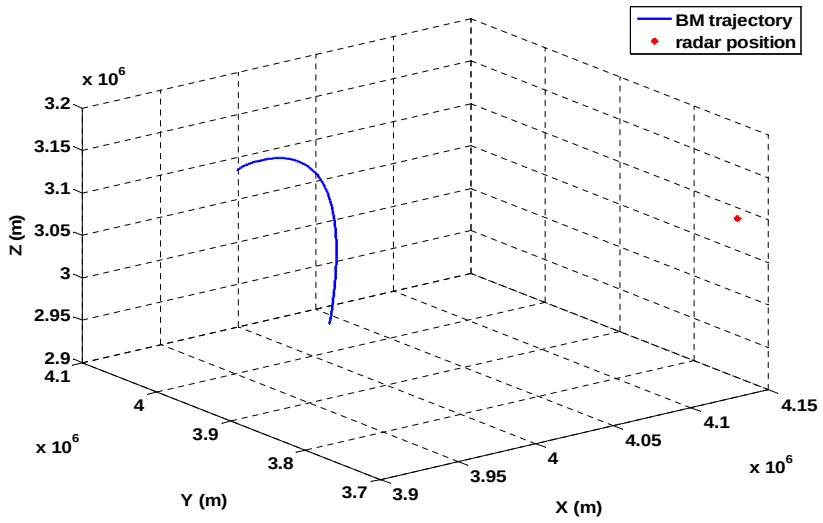


Fig. 5. Simulated BM trajectory and radar position in the ECEF coordinate system

associated uncertainties Σ_0^p and Σ_0^v are given by:

$$\Sigma_0^{\mathbf{p}} = \begin{bmatrix} 100, & 0, & 0 \\ 0, & 100, & 0 \\ 0, & 0, & 100 \end{bmatrix} (m), \quad \Sigma_0^{\mathbf{v}} = \begin{bmatrix} 1, & 0, & 0 \\ 0, & 1, & 0 \\ 0, & 0, & 1 \end{bmatrix} (m/s) \tag{48}$$

The BM parameters n, q and γ are modelled as:

$$\begin{aligned} n_0 &\sim N(\cdot|\overline{n}_0, (\sigma_0^n)^2), \quad \text{with} \;\; \overline{n}_0 = 3, \quad \sigma_0^n = 0.1 \\ q_0 &\sim N(\cdot|\overline{q}_0, (\sigma_0^q)^2), \quad \text{with} \;\; \overline{q}_0 = 0.01, \quad \sigma_0^q = 0.001 \\ \gamma_0 &\sim N(\cdot|\overline{\gamma}_0, (\sigma_0^\gamma)^2), \quad \text{with} \;\; \overline{\gamma}_0 = 2 \cdot 10^{-4}, \quad \sigma_0^\gamma = 10^{-5} \end{aligned} \tag{49}$$

where n_0, q_0 and γ_0 represent BM parameters at $t = 0$. The means $\overline{n}_0$, $\overline{q}_0$ and $\overline{\gamma}_0$ represent initially detected BM parameter values, and σ_0^n, σ_0^q and σ_0^γ represent the associated standard deviations.

State and measurement models: The uncertainty vectors for three state models (boost (15), coast (18) and reentry (20)) are defined as:

$$\begin{aligned} \mathbf{w}_t^b &\sim N(\cdot|\mathbf{0}_{6\times 1}, diag([1,1,1,(10^{-5})^2,(10^{-1})^2,(10^{-3})^2])) \\ \mathbf{w}_t^c &\sim N(\cdot|\mathbf{0}_{6\times 1}, diag([1,1,1(10^{-8})^2])) \\ \mathbf{w}_t^r &\sim N(\cdot|\mathbf{0}_{6\times 1}, diag([1,1,1,(10^{-5})^2])) \end{aligned} \tag{50}$$

where $diag([a_1, ..., a_n])$ represents an $n \times n$ diagonal matrix with elements on the diagonal line being

$[a_1, ..., a_n]$ and others being zeros.

The state-dependent transition probabilities between different state models are set as:

$$\begin{array}{c} \\ boost \\ coast \\ reentry \end{array} \begin{array}{c} \begin{array}{ccc} boost & coast & reentry \end{array} \\ \begin{pmatrix} 1-p_1(h_t) & p_1(h_t) & 0 \\ 0 & p_2(h_t) & 1-p_2(h_t) \\ 0 & 0 & 1 \end{pmatrix} \end{array} \tag{51}$$

where $p_1(h_t) = CDF(h_t|m_{h1}, \sigma_{h1})$ and $p_2(h_t) = CDF(h_t|m_{h2}, \sigma_{h2})$. Related parameters are set as: $m_{h1} = 35000$ (m), $m_{h2} = 25000$ (m), and $\sigma_{h1} = \sigma_{h2} = 3000$ (m).

The measurement model in (27) uses Gaussian noises as:

$$\mathbf{n}_t^m \sim N(\cdot|\mathbf{0}_{3\times 1}, \Sigma_m) \tag{52}$$

where

$$\Sigma_m = diag([(100)^2\ (m)^2, (0.1)^2\ (rad)^2, (0.1)^2\ (rad)^2]) \tag{53}$$

The aforementioned parameter values have been used in the throughout simulations unless explicitly mentioned to set to other values.

A. Modelling system comparison

In this section, we compare the proposed SD-IMMGPF approach with those using a constant transition probabilities-based multiple modelling system, including four IMM approaches implemented by extended Kalman filter in [14]–[17], unscented Kalman filter, particle filter in [27] and Gaussian particle filter. For convenience, these four approaches are denoted as CTP-IMMEKF, CTP-IMMUKF, CTP-IMMPF and CTP-IMMGPF for short. Comparisons have been made in terms of the flight phases probabilities, position and velocity estimates.

1) Estimation of flight phase probabilities: The estimated probabilities of a particular BM flight phase (boost, coast, reentry) is compared. For the particle filtering-based methods, 10,000 particles are used for the filtering corresponding to every mode (the same number is applied for the following simulations unless otherwise stated).

One hundred Monte Carlo simulations are performed and the averaged flight phases probabilities obtained from different methods are plotted in Fig. 6. From the figure, we can see the advantages of the proposed method over other constant transition probabilities-based ones from two aspects: (i) the

estimated probabilities by the SD-IMMGPF method are better matched with the ground truth and there are no obvious fluctuations for the estimated model probabilities during a particular phase period; and (ii) in the transition periods between different phases, the change of the mode probabilities estimated by the SD-IMMGPF method reacts much faster to the true mode change. The obtained advantages are attributed to the state-dependent transition probabilities between state models corresponding to different flight phases, which reflect the true flight phase transitions of the BM in a more realistic way.

2) Estimation of position and velocity: Secondly, we compare the tracking accuracy for BM positions and velocities by different methods. The root-mean-square-error (RMSE) is used to evaluate the tracking accuracy.

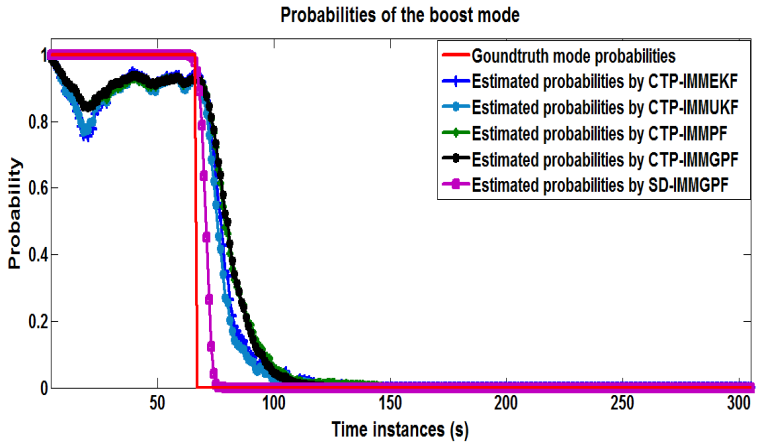
Figures 7 and 8 show the averaged RMSEs from a hundred Monte Carlo simulations for position and velocity at every time instance during particular time intervals, respectively. Besides, the averaged position and velocity RMSEs of these intervals are further given in Tables II and III, from which can see that the advantages (smaller RMSEs) of the proposed SD-IMMGPF approach over other. We need to emphasize that compared with its counterpart of the CTP-IMMGPF approach using the exact same GPF based implementation approach, the proposed SD-IMMGPF approach still achieves better results especially during intervals just after phase transitions, thanks to the better flight phases probabilities estimations during these intervals as shown in Fig. 6 by exploiting the state dependent transition probabilities.

TABLE II
THE AVERAGED POSITION RMSES $\text{RMSES}_{(m)}$ OF DIFFERENT PHASES FOR 100 MONTE CARLO SIMULATIONS.

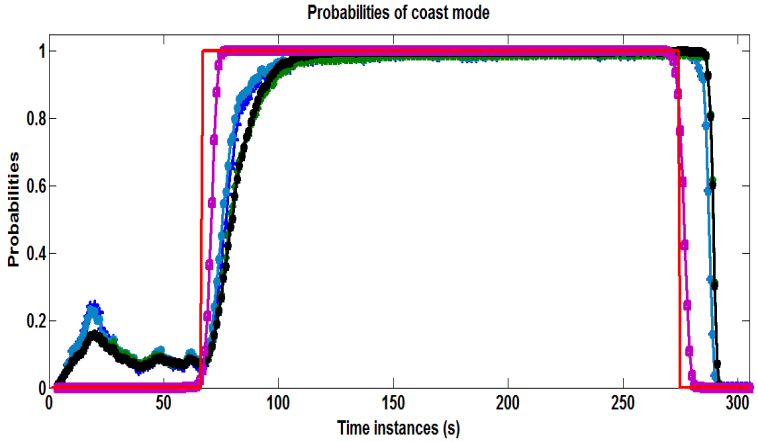
	CTP-IMMEKF	CTP-IMMUKF	CTP-IMMPF	CTP-IMMGPF	SD-IMMGPF
Averaged RMSE for 70-90 (s)	304.88	331.06	292.63	279.13	238.46
Averaged RMSE for 100-200 (s)	173.77	181.76	154.62	155.67	143.07
Averaged RMSE for 280-300 (s)	311.62	298.36	348.44	345.50	227.13

TABLE III
THE AVERAGED VELOCITY $\text{RMSES}_{(m/s)}$ OF DIFFERENT PHASES FOR 100 MONTE CARLO SIMULATIONS.

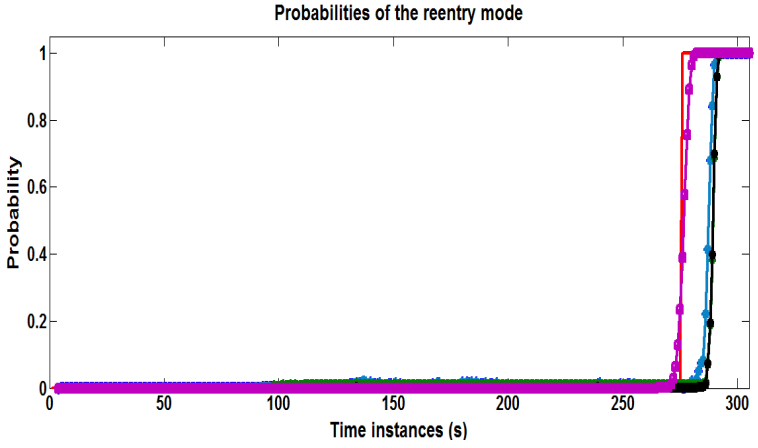
	CTP-IMMEKF	CTP-IMMUKF	CTP-IMMPF	CTP-IMMGPF	SD-IMMGPF
Averaged RMSE for 70-90 (s)	81.24	90.35	75.14	65.52	38.21
Averaged RMSE for 100-200 (s)	13.78	9.41	7.85	7.71	5.09
Averaged RMSE for 280-300 (s)	71.58	76.18	84.66	77.23	55.69



(a) Boost phase

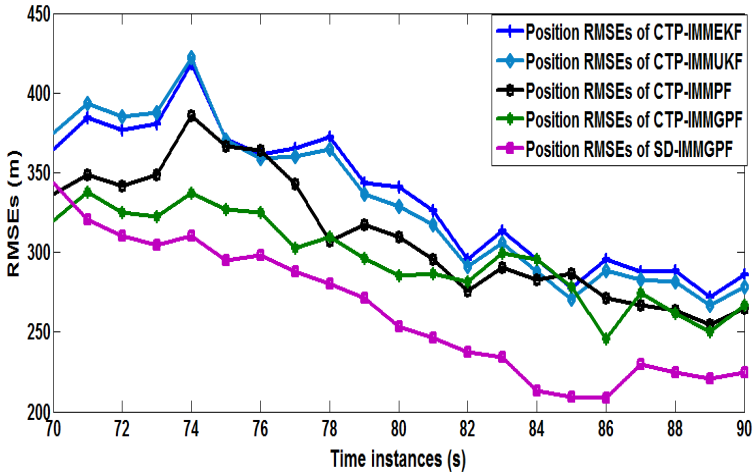


(b) Coast phase

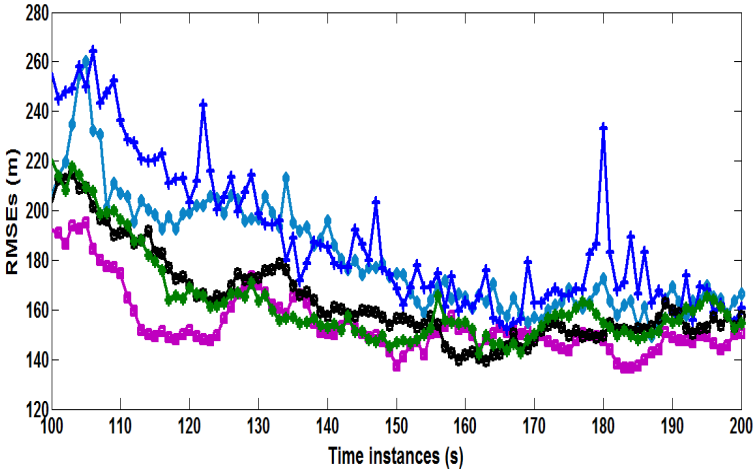


(c) Reentry phase

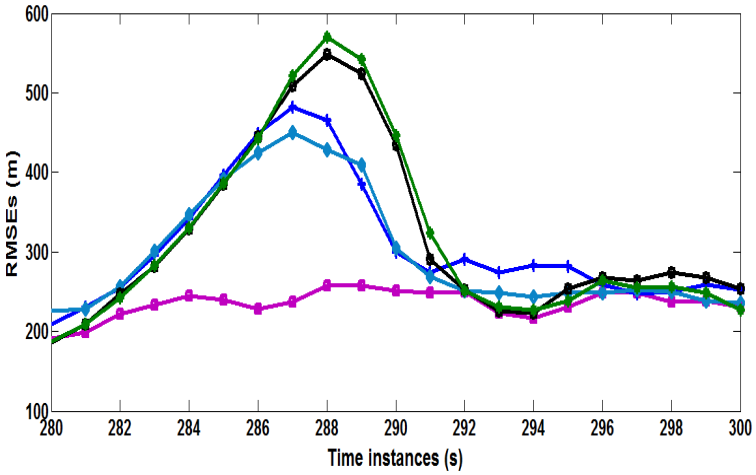
Fig. 6. Estimated flight phases probabilities by different estimation algorithms.



(a) After the transition from boost to coast phase

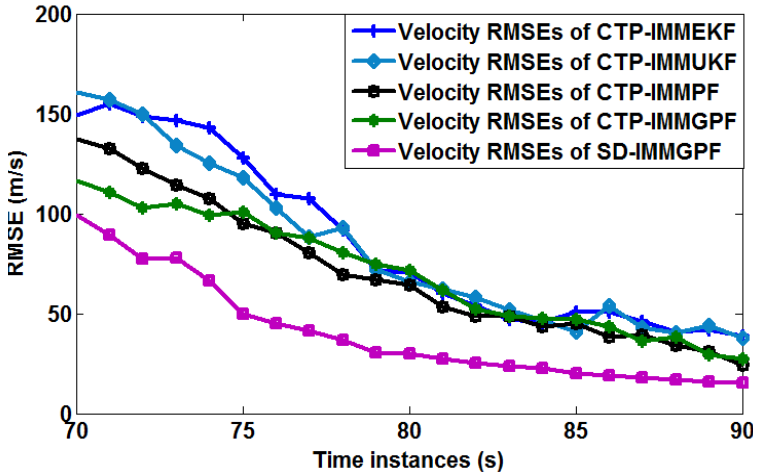


(b) During the coast phase

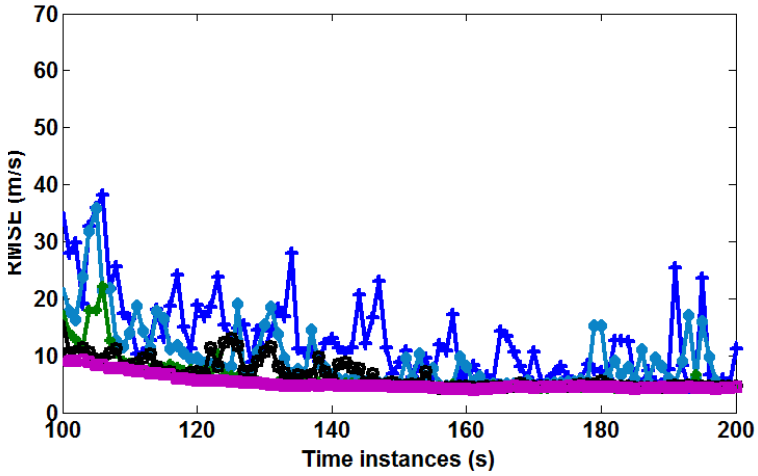


(c) After the transition from coast to entry phase

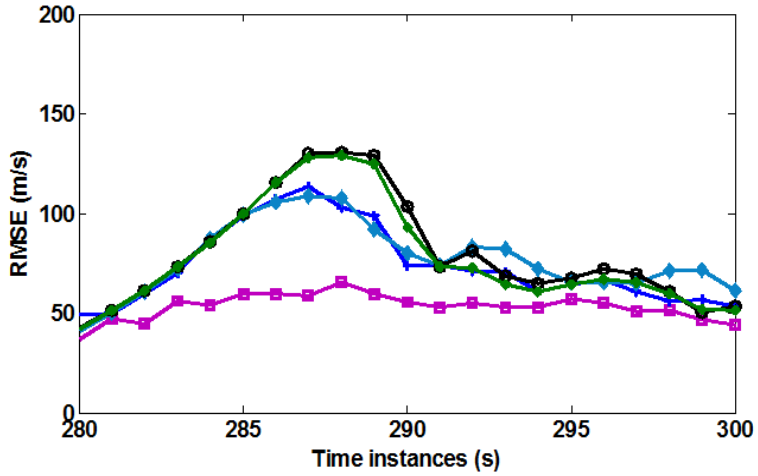
Fig. 7. The position RMSE curves during different intervals.



(a) After the transition from boost to coast phase



(b) During the coast phase



(c) After the transition from coast to entry phase

Fig. 8. The velocity RMSE curves during different intervals.

B. Implementation methods comparisons

We particularly compare two methods of implementing the Bayesian inference: the SD-IMMPF and proposed SD-IMMGPF algorithms, where the same state dependent transition modeling is adopted. Firstly, the two algorithms are evaluated under different number of particles using the aforementioned parameter settings for initialisation, state model and measurement model. Note that the particle filter is a numerical implementation of exact Bayesian estimation which is supposed to be the optimal solution for the problem. When the number of particles are enough large, there is no conservativeness. However, the performance may degrade with the decrease of the number of particles.

One hundred Monte Carlo simulations have been made. The estimated averaged position and velocity RMSEs curves at every time instance are presented in Fig. 9. The averaged position and velocity RMSEs of the related intervals are further given in Tables IV and V.

TABLE IV
COMPARISONS OF THE AVERAGED POSITION RMSEs$_{(m)}$ BETWEEN SD-IMMGPF AND SD-IMMPF .

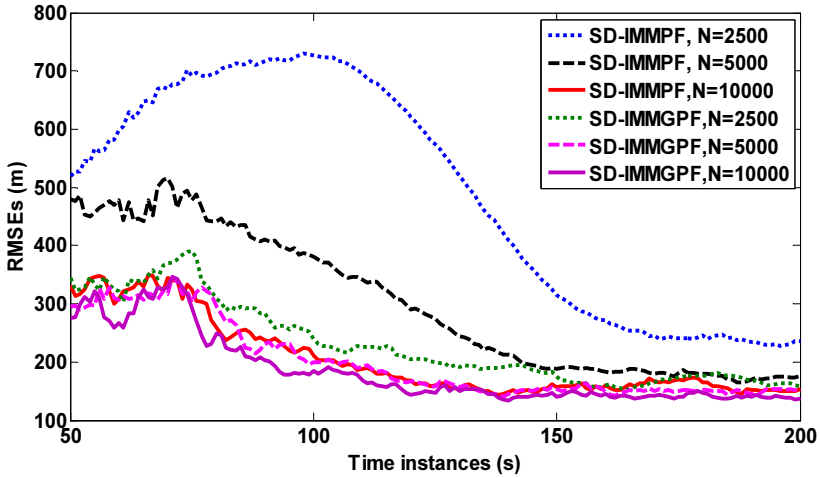
	SD-IMMPF	SD-IMMGPF
N=2500	475.75	231.61
N=5000	302.41	217.54
N=10000	216.81	207.56

TABLE V
COMPARISONS OF THE AVERAGED VELOCITY RMSEs$_{(m/s)}$ BETWEEN SD-IMMGPF AND SD-IMMPF .

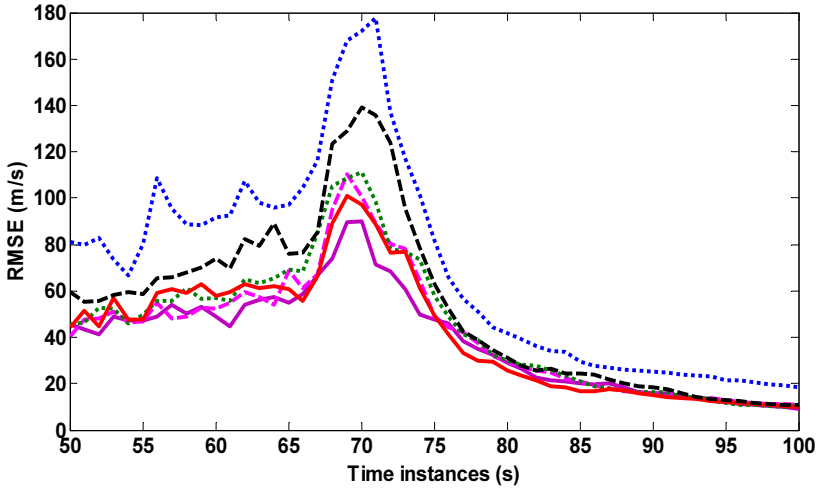
	SD-IMMPF	SD-IMMGPF
N=2500	64.69	51.58
N=5000	57.13	46.49
N=10000	47.27	40.97

From the results, it can be observed that the performance of the SD-IMMPF is heavily affected by the particle size. It becomes worse as the particle number reduces. That is because the SD-IMMPF approach only applies the state model to generate new particles as mentioned in Section III. It is likely to generate more outliers with low likelihood values, thus a comparatively larger number of particles are needed to guarantee good performance.

Secondly, we test different algorithms with different particle sizes under a comparatively worse initial condition with larger uncertainties set for the initial position/velovity components in (54). The RMSE



(a) The position RMSE curves



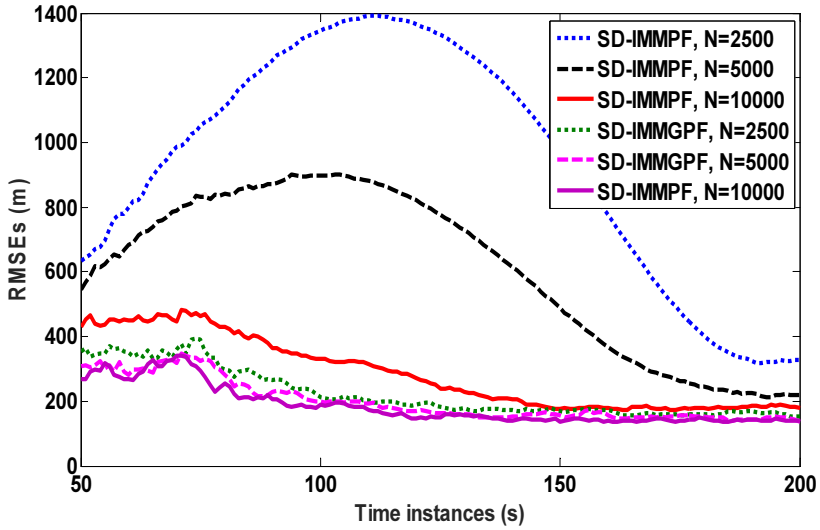
(b) The velocity RMSE curves

Fig. 9. The position RMSE and velocity RMSE curves by the SD-IMMPF and the proposed SD-IMMGPF method.

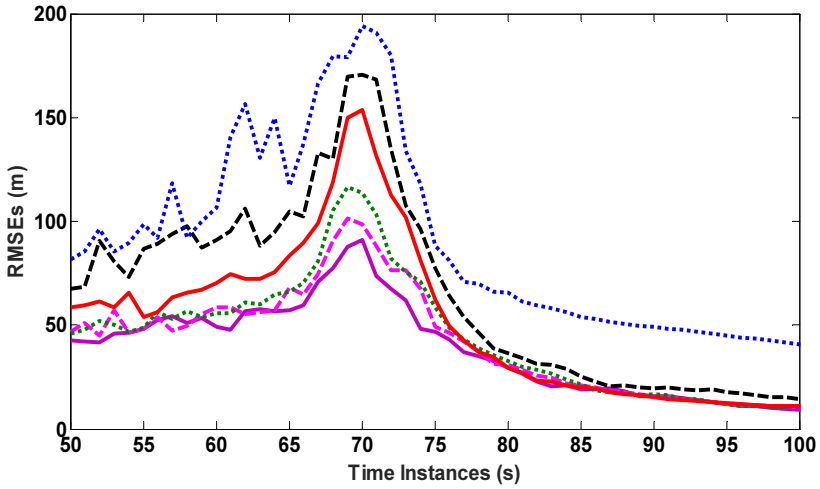
curves and averaged RMSE values during corresponding intervals are shown in Fig. 10 as well as Tables VI and VII.

$$\Sigma_0^{\mathbf{p}} = \begin{bmatrix} 400, & 0, & 0 \\ 0, & 400, & 0 \\ 0, & 0, & 400 \end{bmatrix} (m), \quad \Sigma_0^{\mathbf{v}} = \begin{bmatrix} 4, & 0, & 0 \\ 0, & 4, & 0 \\ 0, & 0, & 4 \end{bmatrix} (m/s) \tag{54}$$

Compared with the previous results, we can see that the performance of the SD-IMMPF is also significantly affected by the initial condition. As the initial condition becomes worse, the performance of the SD-IMMPF becomes worse; however, the proposed SD-IMMGPF is much more robust to the initial



(a) The position RMSE curves



(b) The velocity RMSE curves

Fig. 10. The position RMSE and velocity RMSE curves by the SD-IMMPF and the proposed SD-IMMGPF method under worse initial conditions.

TABLE VI
COMPARISONS OF THE AVERAGED POSITION RMSES $RMSEs_{(m)}$ BETWEEN SD-IMMGPF AND SD-IMMPF UNDER WORSE INITIAL CONDITIONS.

	SD-IMMPF	SD-IMMGPF
N=2500	991.94	231.11
N=5000	578.38	214.58
N=10000	293.47	212.17

TABLE VII
COMPARISONS OF THE AVERAGED VELOCITY RMSES$_{(m/s)}$ BETWEEN SD-IMMGPF AND SD-IMMPF UNDER WORSE INITIAL CONDITIONS.

	SD-IMMPF	SD-IMMGPF
N=2500	112.81	55.40
N=5000	71.25	49.16
N=10000	60.41	44.23

conditions.

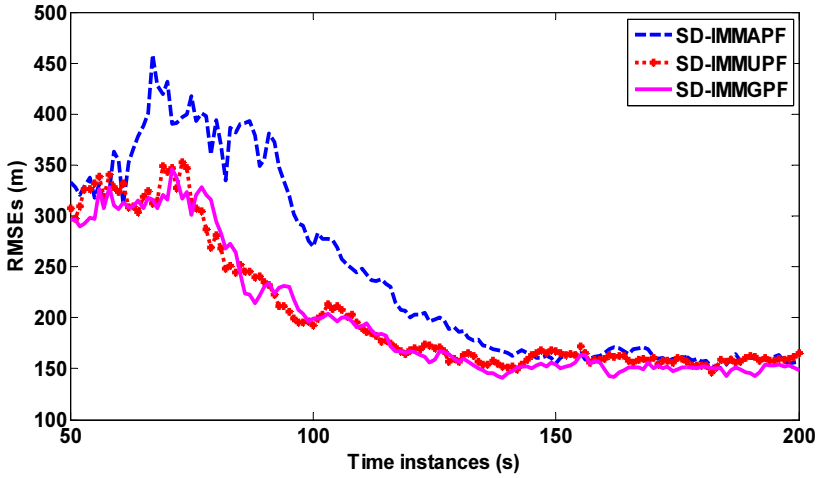
The reason behind it is that, as the initial condition becomes worse, subsequent particles predicted by the state model only will be inconsistent with the true posterior distribution, which leads to the poor performance of the SD-IMMPF. However, in the SD-IMMGPF, the Kalman filtering-based approach is applied to construct importance functions representing a Gaussian approximation of the true posteriori distribution for every mode, from which reasonable particles can still be generated and corrected by the likelihood function. The related performance will not deteriorate too much.

We also investigate various versions of the generic particle filtering (e.g., auxiliary particle filtering [24] and unscented particle filter [25]) for implementing the Bayesian inference of the state-dependent multiple model framework (denoted as SD-IMMAPF and SD-IMMUPF for short). Comparisons are made between SD-IMMAPF, SD-IMMUPF and the proposed method. For a fair comparison, different particle filtering methods follow the same state models, measurement model and initial condition (as given in (47),(48) and (49)).

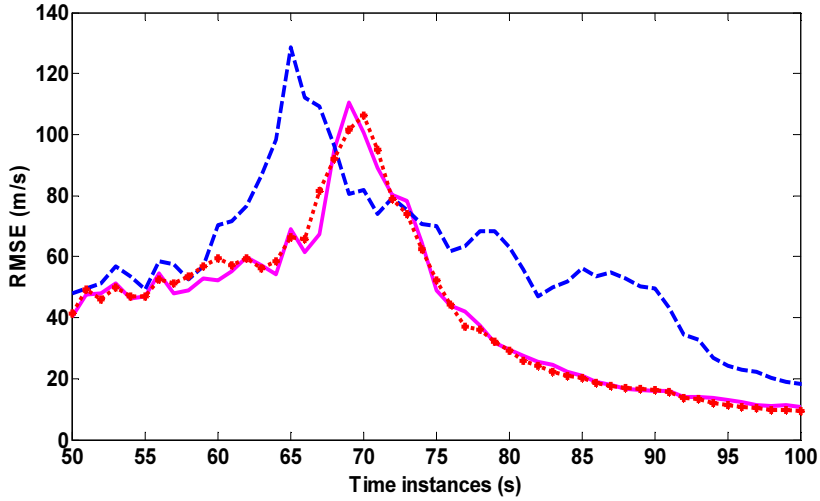
RMSEs at different time instances obtained from 100 Monte Carlo simulations are plotted in Fig. 11, with the averaged position and velocity RMSEs being shown in Table VIII. Besides, the averaged computation time of different filtering algorithms for a single time instance is calculated. We can see that the proposed SD-IMMGPF method achieves smaller RMSEs compared with SD-IMMAPF with a similar computational cost. Although the SD-IMMUPF achieves a similar performance as the SD-IMMGPF, its computational cost is much larger (more than 17 times). Thus, the proposed SD-IMMGPF approach can achieve highly accurate tracking results with a low computational cost.

C. BM Parameters estimation

Finally, the performance in estimating BM parameters is evaluated by comparing with the ground truth values from one hundred Monte Carlo simulations. Based on the initial parameter distributions mentioned



(a) The position RMSE curves



(b) The velocity RMSE curves

Fig. 11. The position and velocity RMSE curves by SD-IMMAPF (N=5000), SD-IMMUPF (N=600) and proposed SD-IMMGPF (N=5000).

TABLE VIII
COMPARISONS OF THE AVERAGED RMSES AND COMPUTATIONAL TIME BETWEEN SD-IMMGPF AND OTHER VERSIONS OF PARTICLE FILTERING BASED IMPLEMENTATION.

	SD-IMMAPF (N=5000)	SD-IMMUPF (N=600)	SD-IMMGPF (N=5000)
Averaged position RMSEs$_{(m)}$	269.59	217.95	217.54
Averaged velocity RMSEs$_{(m/s)}$	59.33	45.91	46.49
Computational time$_{(s)}$	0.07	1.07	0.06

before and SD-IMMGPF filtering, the BM parameter curves for each Monte Carlo run, corresponding mean curve and boundaries determined by three times the standard deviation σ from the mean values, are plotted in Fig. 12. It can be observed that the estimated parameter values quickly converge to the ground truth. In this way, the proposed algorithm can also be used for the parameters estimation, which can be potentially applied for the missile type classification.

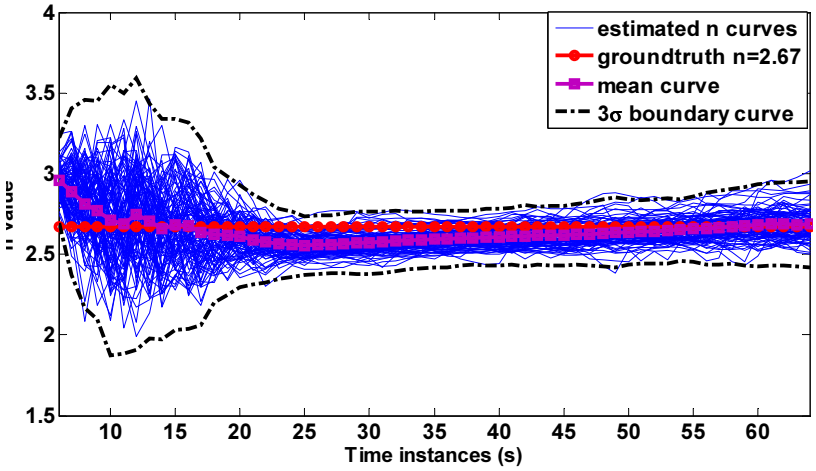
We then evaluated the parameter estimation performance by different filtering methods, by comparing 100 times averaged RMSEs of different parameters at the end of a particular phase (n and q are estimated at the end of the boost phase $t = 64s$ and β is estimated at the end of the reentry phase $t = 305s$). Results for different filtering methods are presented in Table IX. We can see that the proposed SD-IMMGPF method achieves the most accurate parameter estimation results, compared with the CTP-IMMEKF and the SD-IMMPF method.

TABLE IX
THE AVERAGED RMSES OF DIFFERENT PARAMETER ESTIMATIONS FOR 100 MONTE CARLO SIMULATIONS.

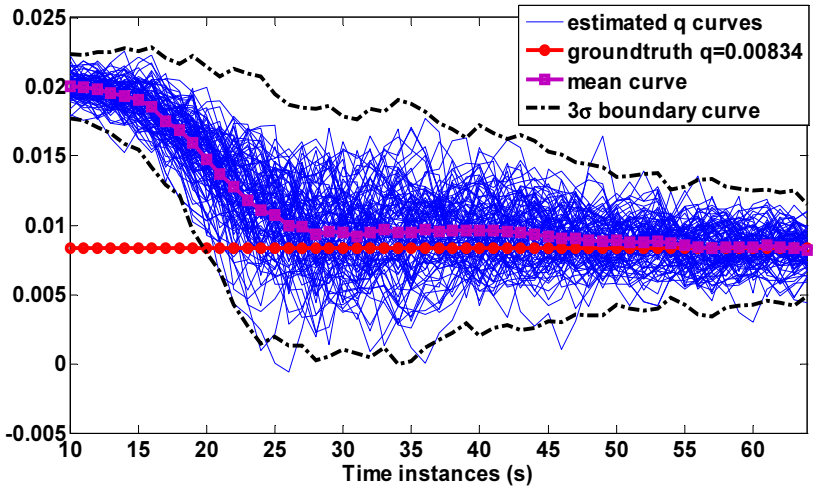
	CTP-IMMEKF	SD-IMMPF	SD-IMMGPF
Averaged RMSE of n ($\times 10^{-2}$)	4.13	3.77	1.20
Averaged RMSE of q ($\times 10^{-4}$)	2.06	2.14	1.94
Averaged RMSE of β	72.70	66.91	48.99

V. CONCLUSIONS AND FUTURE WORK

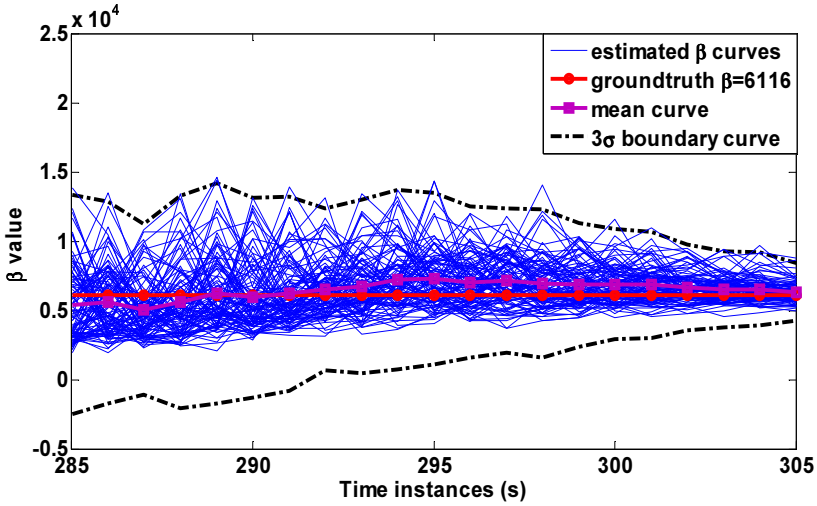
This paper has proposed a new method for tracking the whole trajectory of a ballistic missile. Compared with the current state-of-the-art methods for the ballistic missile tracking, the proposed method has the following novelty both in the state model and Bayesian inference implementation. Firstly, a new modelling framework is applied to model BM movements in different phases. Multiple models are applied to describe the BM dynamics in different phases while transition probabilities between different models are modelled in a state-dependent way rather than fixed values ([14]–[17]). Secondly, a new SD-IMMGPF method is developed to implement the Bayesian inference based on the proposed modelling framework by exploiting both the state model dynamics and measurement information in an efficient way. Comprehensive numerical simulation studies show that the proposed method achieves more accurate mode probabilities, state components and parameters estimations compared with others (such as the traditional IMM based approach [14]–[17]) and different particle filtering based implementation approaches ([19] and [20]). Note that the developed algorithm can also be applied to exploit domain knowledge for tracking



(a) n



(b) q



(c) β

Fig. 12. BM parameter estimation from 100 Monte Carlo simulations.

and behaviour type identification of other objects such as vehicle, ships and pedestrians. In this way, the developed method has the potential to be applied in wider application areas such as the situation awareness in public areas, maritime transport, and autonomous vehicles.

For future work, we will further develop the current algorithm from different aspects. From the modelling aspect, the semi-Markov model [28] will be investigated to model the different manoeuvres (e.g. manoeuvring to evade the interceptor) to accommodate more complex movements of the BM; and a model noise with full rank covariance matrix will also be investigated. From the algorithm development aspect, we will investigate the combination of the state-dependent model switching-based multiple model framework with other filtering techniques to deal with the particle loss problem, such as the particle flow algorithm as in [29] or exploiting various numbers of particles in every mode for filtering. Finally, we will consider a more challenging scenario as in [30] and [31], to track the BM by a sensor-networked system considering the possible network-induced phenomena such as missing/fading measurements, sensor saturations, communication delays, and randomly occurring incomplete information.

REFERENCES

[1] X. Li and V. Jilkov, "Survey of Maneuvering Target Tracking. Part II: Motion Models of Ballistic and Space Targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 1, pp. 96–119, 2010.

[2] M. Hough, "Optimal guidance and nonlinear estimation for interception of accelerating targets," *Journal of Guidance, Control, and Dynamics*, vol. 18, no. 5, pp. 959–968, 1995.

[3] M. Hough, "Improved performance of recursive tracking filters using batch initialization and process noise adaptation," *Journal of Guidance, Control, and Dynamics*, vol. 22, no. 5, pp. 675–681, 1999.

[4] M. Hough, "Nonlinear recursive filter for boost trajectories," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 5, pp. 991–997, 2001.

[5] Y. Li, T. Kirubarajan, Y. Bar-Shalom, and M. Yeddanapudi, "Trajectory and Launch Point Estimation for Ballistic Missiles from Boost Phase LOS Measurements," in *IEEE Aerospace Conference, Snowmass at Aspen, CO, USA*, 1999.

[6] N. Wu, L. Chen, Y. Lei, and F. Meng, "Adaptive estimation algorithm of boost-phase trajectory using binary asynchronous observation," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 230, no. 14, pp. 2661–2672, 2016.

[7] X. He, Y. Bi, and Y. Guo, "Target tracking algorithm of ballistic missile in boost phase based on ground-based radar systems," *Journal of Information and Computational Science*, vol. 12, no. 2, pp. 855–864, 2015.

[8] A. Mukherjee, D. Mukherjee, and A. Sengupta, "Filter design for tracking of ballistic target missile using seeker measurements with time lag," in *2013 International Conference on Signal Processing, Image Processing and Pattern Recognition, Coimbatore, India*, 2013.

[9] C. Chen and D. Zhou, "Study on a new algorithm for tracking ballistic missile in free flight phase," *Advanced Materials Research*, vol. 981, pp. 743–753, 2014.

[10] G. Siouris, G. Chen, and J. Wang, "Tracking an Incoming Ballistic Missile Using an Extended Interval Kalman Filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 1, pp. 232–240, 1997.

[11] M. Bruno and A. Pavlov, "Improved Sequential Monte Carlo Filtering for Ballistic Target Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 3, pp. 1103–1108, 2005.

[12] J. Kim, S. Vaddi, S. Menon, and E. Ohlmeyer, "Comparison Between Nonlinear Filtering Techniques for Spiraling Ballistic Missile State Estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 313–328, 2012.

[13] A. Benavoli, L. Chisci, and A. Farina, "Tracking of a Ballistic Missile with A-Priori Information," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 1000–1016, 2007.

[14] R. Cooperman, "Tactical ballistic missile tracking using the interacting multiple model algorithm," in *The Fifth International Conference on Information Fusion, Annapolis, MD, USA*, 2002.

[15] W. Farrell, "Tracking of a Ballistic Missile with A-Priori Information," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 2, pp. 418–426, 2008.

[16] J. Jung and D. Hwang, "The novel impact point prediction of a ballistic target with interacting multiple models," in *13th International Conference on Control, Automation and Systems, Gwangju, Korea*, 2013.

[17] S. Blackman and R. Popoli, "Design and analysis of modern tracking systems," *Artech House Publishers*, 1999.

[18] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting Multiple Model Methods in Target Tracking: A Survey," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 1, pp. 103–123, 1998.

[19] H. Blom and E. Bloem, "Exact Bayesian and Particle Filtering of Stochastic Hybrid Systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 1, pp. 55–70, 2007.

[20] C. Liu, B. Li, and WH. Chen, "Road netwoek based vehicle navigation using an improved IMM particle filter," *IFAC Proceedings Volumes*, vol. 46, no. 10, pp. 193–198, 2013.

[21] J. Kotecha and P. Djuric, "Gaussian Particle Filtering," *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2592–2601, 2003.

[22] R. Bate, D. Mueller, and E.White, "Fundamentals of astrodynamics," *New York: Dover*, 1971.

[23] R. Burden and J. Faires, "Numerical analysis (5th ed.)," *PWS Publishing, Boston, USA*, 1993.

[24] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*, Norwood, MA: Artech House, 2004.

[25] R. Merve, A. Doucet, N. Freitas, and E. Wan, "The unscented particle filter," *Proceedings of the 13th International Conference on Neural Information Processing Systems*, pp. 563–569, 2000.

[26] "https://www.gov.uk/government/publications/cde-themed-competition-defence-against-airborne-threats," *accessed in September, 2015*.

[27] Y. Boers and J. Driessen, "Interacting multiple model particle filter," *IEE Procceeding of Radar, Sonar and Navigation*, vol. 150, no. 5, pp. 344–349, 2003.

[28] X. 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, 2004.

[29] F. Daum and J. Huang, "Particle flow for nonlinear filters," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic*, 2011.

[30] J. Hu, Z. Wang, D. Chen, and F. Alsaadi, "Estimation, filtering and fusion for networked systems with network-induced phenomena: New progress and prospects," *Information Fusion*, vol. 31, pp. 65–75, 2016.

[31] J. Hu, Z. Wang, S. Liu, and H. Gao, "A variance-constrained approach to recursive state estimation for time-varying complex networks with missing measurements," *Automatica*, vol. 64, pp. 155–162, 2016.