Adaptive Kernel Kalman Filter based Belief Propagation Algorithm for Maneuvering Multi-target Tracking

Mengwei Sun, *Member, IEEE*, Mike E. Davies, *Fellow, IEEE*, Ian K. Proudler, James R. Hopgood, *Member, IEEE*

This letter incorporates the adaptive kernel Kalman filter (AKKF) into the belief propagation (BP) algorithm for multi-target tracking (MTT) in single-sensor systems. The algorithm is capable of tracking an unknown and time-varying number of targets, in the presence of false alarms, clutter and measurement-to-target association uncertainty. Experiment results reveal that the proposed method has a favourable tracking performance using the generalized optimal sub-patten assignment (GOSAP) metrics at substantially less computation cost than the particle filter (PF) based MTT BP algorithm.

***Index Terms*—Adaptive kernel Kalman filter, Belief propagation, Data association, Multi-target tracking**

I. INTRODUCTION

The detection and tracking of multiple targets is a fundamental task in a wide variety of sensor-based applications, such as radar, sonar, and navigation [1]. Multi-target tracking (MTT) aims to estimate the hidden states, e.g. position and velocity, of an unknown and time-varying number of targets, using available measurements with considerations of false alarm, missed detection, and data association uncertainty. Commonly used methods for MTT include the joint probabilistic data association filter (JPDAF) [2], the probability hypothesis density (PHD)/cardinalized PHD (CPHD) filters [3], the multi-Bernoulli (MB) filter and its variants [4]. To improve the scalability of MTT in distributed and modular sensor networks, the belief propagation (BP) based data association (DA) algorithm was presented in [5], [6] by introducing a set of data association variables. Specifically, a factor graph is used to represent the factorization of the joint posterior probability density function (pdf) of the target states and association variables, and then the BP is applied to approximate the marginal association probabilities. This algorithm has the nice property that it estimates each single target state individually.

The Kalman filter (KF) and the least mean square (LMS) adaptive filter are commonly used in statistical signal processing applications [7]. The KF provides the optimal Bayesian estimator for linear dynamic state-space models (DSSMs) and the LMS with its variants, e.g. normalized least mean square (NLMS) [8], [9] are recursive solutions to the optimal Wiener filtering problem. The fundamental principle of the KF update step has similarities to the adaptive LMS algorithm [7]. For non-linear and non-Gaussian systems, the KF is not applicable: the extended Kalman filter (EKF) [10] and the unscented Kalman filter (UKF) [11] were proposed but divergence can occur in some non-linear problems [12] [13]. A more general solution to approximate the integrals of the marginal association probabilities and the marginal posterior pdfs of the states in MTT is using a particle-based implementation [6]. Resampling is a necessary step of the particle-based implementation to avoid degeneracy effects but induces an issue of parallelization. In [14]–[16], the adaptive kernel Kalman filter (AKKF) was proposed by using kernel mean embeddings (KMEs) [17] within full model based filters. The AKKF was demonstrated to have significant improvement of estimation performance compared with other non-linear KFs, reduced computation complexity and avoidance of the resampling compared with most particle filters (PFs).

This letter explores the potential of using the AKKF within MTT systems and demonstrates that the AKKF is a plug-and-play alternative for a PF in MTT systems. Furthermore, the simulations demonstrate the good performance of the AKKF based BP algorithm.

II. BACKGROUND

This section briefly summarizes the statistical formulation of the MTT problem and reviews the framework of the AKKF for single target tracking [14]–[16]. In MTT systems, the augmented state of each target is defined as $\mathbf{y}_{n,k} \triangleq \left[\mathbf{x}_{n,k}^{\mathrm{T}}, r_{n,k}\right]^{\mathrm{T}}$ where $\mathbf{x}_{n,k}$ denotes the hidden state of the k-th target at time n and $r_{n,k}$ represents its existence, $r_{n,k} \in \{0, 1\}$ [6]. The index of each target is represented as $k = 1 : K_n$ and K_n is the total number of existing targets at time n. The survival and birth probabilities are defined as $p_{n,k}^{s} \triangleq p(r_{n,k} = 1 \mid r_{n-1,k} = 1)$ and $p_{n,k}^{b} \triangleq p(r_{n,k} = 1 \mid r_{1:n-1,k} = 0)$, respectively. The joint

M. W. Sun, M. E. Davies and J. R. Hopgood are with Institute of Digital Communications, University of Edinburgh, Edinburgh, EH9 3FG, U.K. E-mail: (msun; mike.davies; james.hopgood)@ed.ac.uk. I. Proudler is with the Centre for Signal & Image Processing (CeSIP), Department of Electronic & Electrical Engineering, University of Strathclyde, Glasgow, G1 1XW, U.K. E-mail: ian.proudler@strath.ac.uk.

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) Grant number EP/S000631/1; and the MOD University Defence Research Collaboration (UDRC) in Signal Processing.

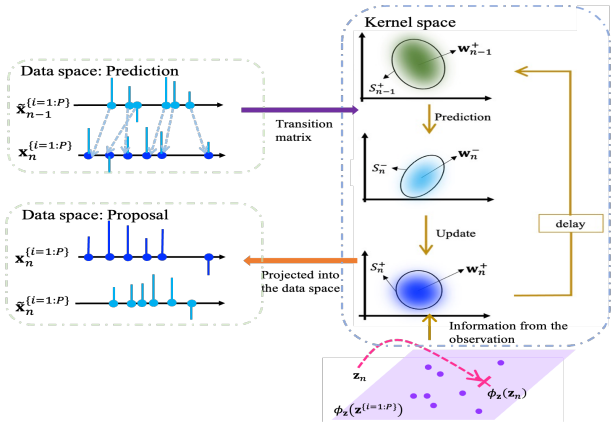

Fig. 1: One iteration of the AKKF. i) Prediction step: the particles $\mathbf{x}_n^{\{1:P\}}$ are calculated by propagating proposal particles $\tilde{\mathbf{x}}_{n-1}^{\{1:P\}}$ through the process function. P represents the number of particles. The KME of the prior distribution is approximated as (1); ii) Update step: the KME of the posterior distribution is approximated as (2); iii) Proposal: generate the proposal particles by accessing the mean and covariance of the hidden state.

state transition pdfs $f(\mathbf{x}_{n,k}, r_{n,k} \mid \mathbf{x}_{1:n-1,k}, r_{1:n-1,k})$ are defined following the Eqn. (5) and Eqn. (6) in [6]. A sensor detects each target with probability $p^d \leq 1$. Clutter occurs according to a Poisson point process with intensity $\lambda_{\text{fa}}(\mathbf{z}_{n,k})$. The measurements vector is denoted as $\mathbf{Z}_n = [\mathbf{z}_{n,1}, \ldots, \mathbf{z}_{n,M_n}]$, where M_n represents the total number of measurements observed at time n. Based on Bayes' theorem, the MTT problem is to calculate the joint posterior pdf of the targets' states $f(\mathcal{Y}_n \mid \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n}, \mathcal{G})$, where $\mathcal{Y}_n = \{\mathbf{y}_{n,1:K_n}\}$ and $\mathcal{G} = \{p_{n,k}^s, p_{n,k}^b, p^d\}$.

The AKKF for single target tracking is formulated by executing the KF in some reproducing kernel Hilbert spaces (RKHSs) [16]. Compared with the EKF which uses the first derivatives to approximated non-linear functions, and the UKF and PF uses sigma points or particles with associated weights in the data space to approximate the pdfs of the hidden state, the AKKF explores the potential of using KMEs in a high or even infinite-dimensional RKHSs such that the non-linear DSSMs become nearly linear. The empirical KME of the prior and posterior pdfs of the hidden states become elements of RKHSs and are represented as linear combinations of some weighted basis vectors as (1) and (2), respectively:

$$f(\mathbf{x}_n|\mathbf{x}_{1:n-1}, \hat{r}_{n-1} = 1, \mathbf{Z}_{1:n-1}) \rightarrow \hat{\mu}_{\mathbf{x}_n}^- = \Phi_n \mathbf{w}_n^-, \tag{1}$$

$$f(\mathbf{x}_n|\mathbf{x}_{1:n-1}, \hat{r}_n = 1, \mathbf{Z}_{1:n}) \rightarrow \hat{\mu}_{\mathbf{x}_n}^+ = \Phi_n \mathbf{w}_n^+, \tag{2}$$

where Φ_n represents the set of the feature mapping vectors embedded by the state particles $\mathbf{x}_n^{\{1:P\}}$, which contains more features than the state particles and can transform and store the state particles more efficiently. The linearity means that the mean vector and covariance matrix of the corresponding KME weights $\mathbf{w}_n^{\mp}$ are predicted and updated in the RKHSs in the KF manner. As illustrated in Fig. 1, the AKKF is executed in three steps: i) The AKKF adaptively propagates particles in a similar manner to the UKF in the data space, and the mean vector and covariance matrix of the corresponding KME weights are predicted with the transition matrix; ii) The mean vector and covariance matrix of the KME weights are updated by calculating the kernel Kalman gain in the RKHSs. The derivation of the kernel Kalman gain is formulated to producing the LMS of the posteriori KME error following the Eqn. (57) – Eqn. (61) in [16]; iii) The proposal particles are drawn according to the importance distribution accessed from $\hat{\mu}_{\mathbf{x}_n}^+$ in the data space, i.e. $\tilde{\mathbf{x}}_n^{\{1:P\}} \sim \mathcal{N}(\mathbb{E}(\mathbf{x}_n), \text{Cov}(\mathbf{x}_n))$.

Algorithm 1 AKKF based BP

Require: Motion models $f_k(\cdot)$, measurement models $h_k(\cdot)$.

1: **Initialization**: Set the initial particles $\mathbf{x}_{0,k}^{\{i=1:P\}} \sim P_{\text{init},k}$, $\Phi_{0,k} := \left[\phi_{\mathbf{x}}(\mathbf{x}_{0,k}^{\{1\}}), \ldots, \phi_{\mathbf{x}}(\mathbf{x}_{0,k}^{\{P\}})\right]$, $\mathbf{w}_{0,k} = 1/P\,[1, \ldots, 1]^{\text{T}}$.
2: **for** $n = 1 : N$ **do**
3: Prediction for $k = 1 : \hat{K}_{n-1}$:
· In the data space, $\mathbf{x}_{n,k}^{\{i\}} = f_k(\tilde{\mathbf{x}}_{n-1,k}^{\{i\}}, \mathbf{u}_{n-1,k}^{\{i\}})$, $i = 1 \ldots P$.
· In the RKHS, execute (4) and (5).
4: Data association: run the BP and calculate the marginal probabilities using Eqn.(22) and Eqn.(23) from [5], and (6) and (7).
5: Update for $k = 1 : \hat{K}_{n-1}$:
· In the data space: $\mathbf{y}_{n,k}^{\{i\}} = h_k(\mathbf{x}_{n,k}^{\{i\}}, \mathbf{v}_{n,k}^{\{i\}})$, $i = 1 \ldots P$. · In the RKHS, execute (9) and (10).
6: Update for $k = \hat{K}_{n-1} + 1 : \hat{K}_{n-1} + \hat{K}_{n,b}$:
· In the data space: $\mathbf{x}_{n,k}^{1:P} \sim f_b(\mathbf{x}_{n,k})$.
· In the RKHS, execute (9) and (10).
7: Proposal particles draw:
$\tilde{\mathbf{x}}_{n,k}^{\{i=1:P\}} \sim \mathcal{N}(\mathbb{E}(\mathbf{x}_{n,k}), \text{Cov}(\mathbf{x}_{n,k}))$.
8: **end for**

III. AKKF BASED BP

The previously proposed AKKF is a single target tracking algorithm. To deal with the MTT problem, the AKKF is plugged into the BP algorithm and the flow diagram of the proposed algorithm is shown in Fig. 2. The algorithm includes three modules, i.e. prediction, data association (DA) and update, which are discussed in the following subsections and its implementation is summarized in Algorithm 1.

A. Prediction of Potential Targets

After the estimation at time $n-1$, the estimates of the total number of the existing potential targets and the corresponding states are denoted as $\hat{K}_{n-1}$ and $\hat{\mathbf{x}}_{n-1,1:\hat{K}_{n-1}}$, respectively. The predictive pdf of each potential target from time $n-1$ to time n is embedded into a RKHS as a linear algebra operation,

$$\begin{aligned} &f(\mathbf{x}_{n,k} \mid \mathbf{x}_{n-1,k}, \hat{r}_{n-1,k} = 1, \mathbf{Z}_{1:n-1}) \\ &\rightarrow \hat{\mu}_{\mathbf{x}_{n,k}}^- = \Phi_{n,k} \mathbf{w}_{n,k}^- = \Phi_{n,k} \left(K_{\tilde{\mathbf{x}}\tilde{\mathbf{x}},k} + \lambda_{\tilde{K}} I\right)^{-1} K_{\tilde{\mathbf{x}}\mathbf{x},k} \mathbf{w}_{n-1,k}^+. \end{aligned} \tag{3}$$

Here, $K_{\tilde{\mathbf{x}}\tilde{\mathbf{x}},k}$ and $K_{\tilde{\mathbf{x}}\mathbf{x},k}$ are Gram matrices. $\Phi_{n-1,k} = \left[\phi_{\mathbf{x}}(\mathbf{x}_{n-1,k}^{\{1\}}), \ldots, \phi_{\mathbf{x}}(\mathbf{x}_{n-1,k}^{\{P\}})\right]$, $\Psi_{n-1,k} = \left[\phi_{\mathbf{x}}(\tilde{\mathbf{x}}_{n-1,k}^{\{1\}}), \ldots, \phi_{\mathbf{x}}(\tilde{\mathbf{x}}_{n-1,k}^{\{P\}})\right]$ and $\Phi_{n,k} = \left[\phi_{\mathbf{x}}(\mathbf{x}_{n,k}^{\{1\}}), \ldots, \phi_{\mathbf{x}}(\mathbf{x}_{n,k}^{\{P\}})\right]$ represent the feature mappings embedded by the preceding particles $\mathbf{x}_{n-1,k}^{\{i=1:P\}}$, the preced-

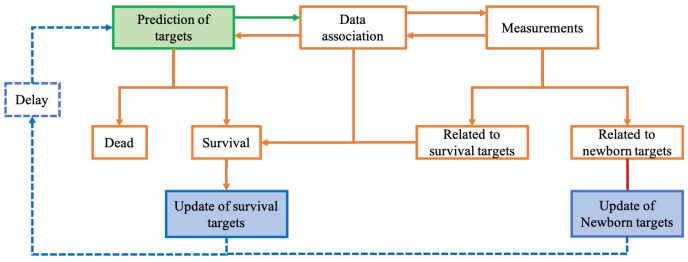

Fig. 2: Flow diagram of the proposed algorithm. The shading boxes are the steps using the AKKF to do the prediction and update. i) Prediction step in the green block follows the prediction step of the AKKF; ii) DA step in the orange blocks; iii) Update step in the blue blocks: the posterior pdfs of the survivors and newborn targets are updated based on the update step of the AKKF.

ing proposal particles $\tilde{\mathbf{x}}_{n-1,k}^{\{i=1:P\}}$, and the current particles $\mathbf{x}_{n,k}^{\{i=1:P\}}$, respectively. $\lambda_{\tilde{K}}$ is the regularization parameter to stabilize the inverse of $K_{\tilde{\mathbf{x}}\tilde{\mathbf{x}},k}$. $\mathbf{w}_{n-1,k}^{+}$ and $\mathbf{w}_{n,k}^{-}$ represent the associated proposal and predictive KME weight vectors of the k-th target, respectively. Based on the derivations in [14], the prior KME weight mean vector $\mathbf{w}_{n,k}^{-}$ in (3) and the covariance matrix $S_{n,k}^{-}$ are calculated as

$$\mathbf{w}_{n,k}^{-} = \left(K_{\tilde{\mathbf{x}}\tilde{\mathbf{x}},k} + \lambda_{\tilde{K}} I\right)^{-1} K_{\tilde{\mathbf{x}}\mathbf{x},k} \mathbf{w}_{n-1,k}^{+} = \Gamma_{n-1,k} \mathbf{w}_{n-1,k}^{+}, \tag{4}$$

$$S_{n,k}^{-} = \Gamma_{n-1,k} S_{n-1,k}^{+} \Gamma_{n-1,k}^{\mathrm{T}} + V_{n,k}, \tag{5}$$

where $\Gamma_{n-1,k}$ represents the change of sample representation from $\Phi_{n-1,k}$ to $\Psi_{n-1,k}$ and $V_{n,k}$ represents the transition residual matrix. $S_{n,k}^{-}$ will be used to calculate the posterior KME weight mean vector and covariance matrix in (9) and (10).

B. Data Association

The marginal posterior pdfs of the hidden states can be obtained by running iterative BP on the factor graph as illustrated in Fig. 3. Following the BP algorithm in [5], [6], the association between measurements and targets is described by two DA vectors which are defined as

- $a_{n,k} \triangleq m \in \{1, \ldots, M_n\}$, the k-th target generates the m-th measurement.
- $a_{n,k} \triangleq m = 0$, missed detection or death of target occurs.
- $b_{n,m} \triangleq k \in \{1, \ldots, \hat{K}_{n-1}\}$, the m-th measurement is hypothesized to be associated with k-th target.
- $b_{n,m} \triangleq k = 0$, clutter or newborn of target occurs.

The approximations of the marginal association probabilities, $p(a_{n,k} = m \mid \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n}, \mathcal{G})$ and $p(b_{n,m} = k \mid \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n}, \mathcal{G})$, are calculated following the Eqn. (22) and Eqn. (23) in [5], respectively. In contrast to the factor graph in [5], [6], two different sets of variable nodes, $r_{n,1:\hat{K}_{n-1}}$ and $\gamma_{n,1:M_n}$, are stretched in Fig. 3, which are defined as: $r_{n,k} = 0/1$ if the k-th potential target is determined to be a survivor/dead; $\gamma_{n,m} = 0/1$ if the m-th measurement is determined to be associated with a survivor or a newborn target. The approximated marginal probabilities of $r_{n,k}$ and $\gamma_{n,m}$ are calculated as (6) and (7), respectively. Note that, since the number of targets is unknown and time-variant, we classify the miss-detection and re-detection of a particular target as

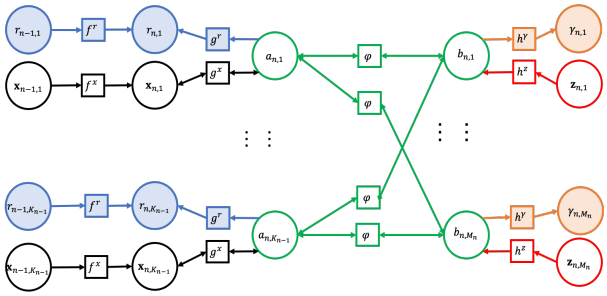

Fig. 3: Graphical model for data association. Variable nodes represented by circles include: the state nodes $\mathbf{x}_{n-1:n,1:\hat{K}_{n-1}}$ in black, the measurement nodes $\mathbf{z}_{n,1:M_n}$ in red, the stretching association nodes $a_{n,1:\hat{K}_{n-1}}$ and $b_{n,1:M_n}$ in green, the existence of target variable nodes $r_{n-1:n,1:\hat{K}_{n-1}}$ in blue and the newborn target associated measurement variable nodes $\gamma_{n,1:M_n}$ in orange. The factor nodes represented by squares are defined as: $f^r \triangleq f(r_{n,k} \mid r_{n-1,k})$, $f^x \triangleq f(\mathbf{x}_{n,k} \mid \mathbf{x}_{n-1,k})$, $g^r \triangleq g^r(r_{n,k} \mid a_{n,k})$, $g^x \triangleq g^x(\mathbf{x}_{n,k}, a_{n,k}; Z_n)$, $h^\gamma \triangleq h^\gamma(\gamma_{n,m} \mid b_{n,m})$, $h^z \triangleq h^z(b_{n,m} \mid \mathbf{z}_{n,m})$, $\varphi \triangleq \varphi(a_{n,k}, b_{n,m})$. The shading nodes are additional nodes compared to [5], [6].

the death and birth of two separate targets. The numbers of the potential newborn targets and dead targets at time n are estimated as $\hat{K}_{n,b} = \|\hat{\gamma}_{n,1:M_n} = 1\|$ and $\hat{K}_{n,d} = \|\hat{r}_{n,1:\hat{K}_{n-1}} = 0\|$, respectively. Hence, the total number of the existing potential targets at time n is $\hat{K}_n = \hat{K}_{n-1} + \hat{K}_{n,b} - \hat{K}_{n,d}$.

$$\hat{r}_{n,k} = \max_{r_{n,k} \in \{0,1\}} \left[p(r_{n,k} \mid a_{n,k}, \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n}, \mathcal{G})\right], \tag{6}$$

$$\hat{\gamma}_{n,m} = \max_{\gamma_{n,m} \in \{0,1\}} \left[p(\gamma_{n,m} \mid b_{n,m}, \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n}, \mathcal{G})\right]. \tag{7}$$

C. Posterior Estimation of Survivors and Newborn Targets

For the target that is determined to be a survivor, the marginal posterior pdf of the hidden state, $f(\mathbf{x}_{n,k} \mid \hat{r}_{n,k} = 1, a_{n,k}, \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n-1}, \mathcal{G})$, is derived as

$$\begin{aligned} &f(\mathbf{x}_{n,k} \mid \hat{r}_{n,k} = 1, \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n-1}, \mathcal{G}) \\ &= \sum_{m=1}^{M_n} f(\mathbf{x}_{n,k} \mid \hat{r}_{n,k} = 1, a_{n,k} = m, \mathbf{z}_{n,m}, \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n-1}, \mathcal{G}) \\ &\quad \times p(a_{n,k} = m \mid \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n}, \mathcal{G}). \end{aligned} \tag{8}$$

Here, the pdf $f(\mathbf{x}_{n,k} \mid \hat{r}_{n,k} = 1, a_{n,k} = m, \mathbf{z}_{n,m}, \mathcal{Y}_{0:n-1}, \mathbf{Z}_{1:n-1}, \mathcal{G})$ in equation (8), is embedded into the RKHS as $\Phi_{n,k}\mathbf{w}_{n,k}^{+}$ following the update step of the AKKF, where the posterior KME weight mean vector $\mathbf{w}_{n,k}^{+}$ and covariance matrix $S_{n,k}^{+}$ are updated as [14], [15]:

$$\mathbf{w}_{n,k}^{+} = \mathbf{w}_{n,k}^{-} + \left[S_{n,k}^{-}\left(G_{\mathbf{zz},k} S_{n,k}^{-} + \kappa I\right)^{-1}\right]\left(G_{:,\mathbf{z}_{n,m}} - G_{\mathbf{zz},k}\mathbf{w}_{n,k}^{-}\right) \tag{9}$$

$$S_{n,k}^{+} = S_{n,k}^{-} - \left[S_{n,k}^{-}\left(G_{\mathbf{zz},k} S_{n,k}^{-} + \kappa I\right)^{-1}\right] G_{\mathbf{zz},k} S_{n}^{-}. \tag{10}$$

Here, the kernel vector of the measurement is $G_{:,\mathbf{z}_{n,m}} = \Upsilon_{n,k}^{\mathrm{T}} \phi(\mathbf{z}_{n,m})$, $G_{\mathbf{zz},k}$ represents the Gram matrix of the measurements, and the feature mappings of the measurement particles corresponding to the k-th target is $\Upsilon_{n,k} = \left[\phi_{\mathbf{z}}(\mathbf{z}_{n,k}^{\{1\}}), \ldots, \phi_{\mathbf{z}}(\mathbf{z}_{n,k}^{\{P\}})\right]$. Then, the mean and covariance information, $\mathbb{E}(X_{n,k})$ and $\mathrm{Cov}(X_{n,k})$, are extracted from $\hat{\mu}_{\mathbf{x}_{n,k}}^{+}$ and passed to the data space for the proposal particles generation as $\tilde{\mathbf{x}}_{n,k}^{\{i=1:P\}} \sim \mathcal{N}\left(\mathbb{E}(\mathbf{x}_{n,k}), \mathrm{Cov}(\mathbf{x}_{n,k})\right)$ [14].

The state particles of each potential newborn target are drawn following the prior distribution, i.e. $\mathbf{x}_{n,k}^{1:P} \sim f_b(\mathbf{x}_{n,k})$, where $k = \hat{K}_{n-1}+1 : \hat{K}_{n-1}+\hat{K}_{n,b}$. The posterior KME weight mean vector and its covariance matrix are updated according to (9) and (10) where the prior KME weight mean vector and covariance matrix are set as $\mathbf{w}_{n,k}^{-} = \frac{1}{P}[1,\ldots,1]^{\mathrm{T}}$ and $S_{n,k}^{-} = I_{P\times P}$. In this letter, all the measurements at time n are determined to be associated with newborn targets estimated as $\hat{\mathbf{x}}_{n,k}, k = \hat{K}_{n-1}+1 : \hat{K}_{n-1}+\hat{K}_{n,b}$ if $\gamma_{n,m} = 1$, but will be pruned if it is determined to be dead at the next time slot.

IV. SIMULATION RESULTS

The motion model of the hidden states is

$$\mathbf{x}_{n,k} = \begin{cases} \mathbf{F}\mathbf{x}_{n-1,k} + \mathbf{G}\mathbf{u}_{n-1,k}, & \text{if} \quad (r_{n,k}, r_{1:n-1,k}) = (1,1) \\ \emptyset, & \text{if} \quad (r_{n,k}, r_{1:n-1,k}) = (0,1) \\ \mathbf{x}_{0,k} \sim \mathcal{N}(\bar{\mathbf{x}}_{0,k}, \mathbf{P}_{0,k}), & \text{if} \quad (r_{n,k}, r_{1:n-1,k}) = (1,0). \end{cases} \tag{11}$$

Here, the hidden states are $\mathbf{x}_{n,k} = [\xi_{n,k}, \dot{\xi}_{n,k}, \eta_{n,k}, \dot{\eta}_{n,k}]^T$, where $(\xi_{n,k}, \eta_{n,k})$ and $(\dot{\xi}_{n,k}, \dot{\eta}_{n,k})$ represent the target position and the corresponding velocity in X-axis and Y-axis, respectively. The process matrices $\mathbf{F}$ and $\mathbf{G}$ are specified following [14]. The process noise is a 2×1 vector, i.e. $\mathbf{u}_{n,k} = \left[u_x, u_y\right]_{n,k}^{\mathrm{T}}$, which follows Gaussian distribution $\mathbf{u}_{n,k} \sim \mathcal{N}(\mathbf{0}, \sigma_u^2\mathbf{I}_2)$, $\sigma_u = 0.01$ and $\mathbf{I}_2$ is the 2×2 identity matrix. The survival probability is $p^s = 0.98$, the birth probability is $p^b = 0.3$, and the number of newborn targets is a Poisson distribution with mean $\mu_b = 1$. A single sensor system is considered with a measurement range of 300m $\times$ 300m. The measurement model (range and bearing) is defined as

$$\mathbf{z}_{n,k} = \begin{cases} \begin{bmatrix} \sqrt{\xi_{n,k}^2 + \eta_{n,k}^2} \\ \tan^{-1}(\xi_{n,k}, \eta_{n,k}) \end{bmatrix} + \mathbf{v}_{n,k}, & \text{detection} \\ \mathbf{v}_{n,k}, & \text{missed detection} \end{cases} \tag{12}$$

where the inverse tangent is the four-quadrant inverse tangent function and $\mathbf{v}_{n,k}$ is the measurement noise.

The performance comparisons in single target tracking are presented in [14], [16], from which we can see that divergence of the UKF is much worse than the AKKF and PFs when the measurement model is non-linear. Hence, the PF is chosen as the comparison to validate the proposed algorithm. Fig. 4 and Fig. 5 display the comparisons of the tracking performance and the estimated number of targets, respectively, which are obtained by two BP-based filters: the PF and the quadratic kernel AKKF. The number of particles used is 50. It can be seen from Fig. 4 that with a small number of particles, missed-detections happen slightly more often for the PF based BP. Fig. 6 shows the average generalized optimal sub-patten assignment (GOSAP) and localization estimation error of detected targets with the correspond running time obtained for 100 random realizations. The number of particles for the AKKF are set as $P = [20, 50, 100]$, and the number of particles for the PF are set as $P = [20, 50, 100, 2000]$. From this figure, we can conclude that the computation complexity of the AKKF based BP is very favorable and offers a great scalability for tracking applications in MTT systems.

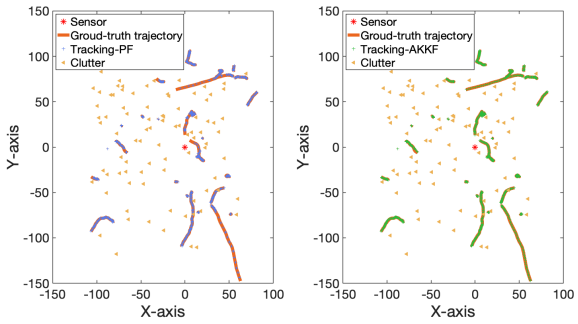

Fig. 4: Tracking performance comparison. Legend: Red $*$ – single sensor; Orange lines – ground truth trajectories; Blue +, estimated trajectories achieved by the PF–BP; Green +, estimated trajectories achieved by the AKKF–BP; Yellow triangles, clutter.

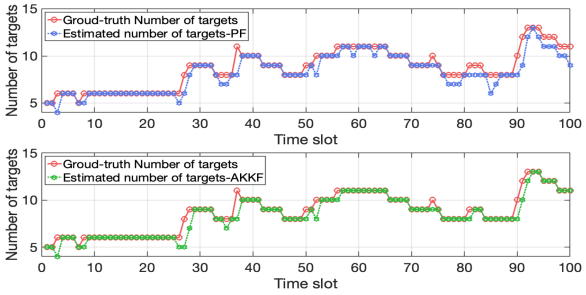

Fig. 5: Number of estimated targets comparison. Legend: Orange lines – ground truth number of targets; Blue +, estimated number of targets achieved by the PF based BP; Green +, estimated number of targets achieved by the AKKF based BP.

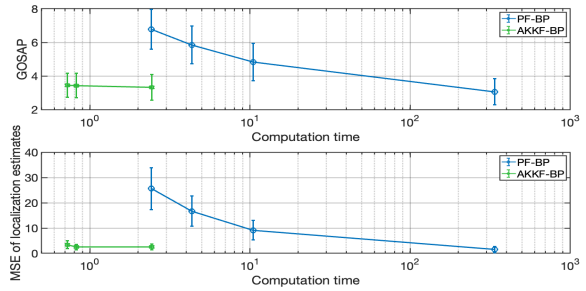

Fig. 6: GOSAP and localization estimation error of the detected targets comparisons.

V. CONCLUSION

In this paper, we showed that the AKKF can replace a PF in the MTT BP algorithm of Williams et. al [5]. This work demonstrates the immediate potential for using the AKKF as a plug-and-play filtering module in a wide-variety of Bayesian filtering applications. The investigation of the AKKF to the optimal Wiener filtering problem, such as the application in adaptive feedback cancellation and acoustic echo cancellation will be reported in a forthcoming publication.

References

[1] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3460–3475, 2013.

[2] D. Musicki and R. Evans, "Joint integrated probabilistic data association: JIPDA," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 40, no. 3, pp. 1093–1099, 2004.

[3] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3553–3567, 2007.

[4] M. Beard, B. T. Vo, and B.-N. Vo, "A solution for large-scale multi-object tracking," *IEEE Trans. Signal Process.*, vol. 68, pp. 2754–2769, 2020.

[5] J. Williams and R. Lau, "Approximate evaluation of marginal association probabilities with belief propagation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 2942–2959, 2014.

[6] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch, "A scalable algorithm for tracking an unknown number of targets using multiple sensors," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3478–3493, 2017.

[7] D. P. Mandic, S. Kanna, and A. G. Constantinides, "On the intrinsic relationship between the least mean square and kalman filters [lecture notes]," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 117–122, 2015.

[8] T. Aboulnasr and K. Mayyas, "A robust variable step-size lms-type algorithm: analysis and simulations," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 631–639, 1997.

[9] H.-C. Shin, A. Sayed, and W.-J. Song, "Variable step-size nlms and affine projection algorithms," *IEEE Signal Processing Letters*, vol. 11, no. 2, pp. 132–135, 2004.

[10] H. W. Sorenson, *Kalman Filtering: Theory and Application*. NJ: IEEE: Piscataway, 1985.

[11] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal Processing, Sensor Fusion, and Target Recognition VI*, I. Kadar, Ed., vol. 3068, International Society for Optics and Photonics. SPIE, 1997, pp. 182 – 193. [Online]. Available: https://doi.org/10.1117/12.280797

[12] K. Ito and K. Xiong, "Gaussian filters for nonlinear filtering problems," *IEEE Trans. Autom. Control.*, vol. 45, pp. 910–927, 2000.

[13] J. Kotecha and P. Djuric, "Gaussian particle filtering," *IEEE Trans. Signal Process.*, vol. 51, no. 10, pp. 2592–2601, 2003.

[14] M. Sun, M. E. Davies, I. Proudler, and J. R. Hopgood, "Adaptive kernel Kalman filter," in *2021 Sensor Signal Processing for Defence Conference (SSPD)*, 2021, pp. 1–5.

[15] M. Sun, M. E. Davies, J. R. Hopgood, and I. Proudler, "Adaptive kernel Kalman filter multi-sensor fusion," in *2021 IEEE 24th International Conference on Information Fusion (FUSION)*, 2021, pp. 1–8.

[16] M. Sun, M. E. Davies, I. K. Proudler, and J. R. Hopgood, "Adaptive kernel Kalman filter," 2022. [Online]. Available: https://arxiv.org/abs/2203.08300

[17] L. Song, K. Fukumizu, and A. Gretton, "Kernel embeddings of conditional distributions: A unified kernel framework for nonparametric inference in graphical models," *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 98–111, Jun. 2013.