Soft-Data-Constrained Multi-Model Particle Filter for Agile Target Tracking

Sepideh Seifzadeh, Bahador Khaleghi and Fakhri Karray
Electrical and Computer Engineering Department
University of Waterloo
Waterloo, Ontario, CANADA
Email: sseifzad, bkhalegh, karray@uwaterloo.ca

Abstract—The performance of Bayesian filtering based methods can be enhanced by using extra information incorporated as specific constraints into the filtering process. Following the same principle, this paper proposes a Soft-Data-Constrained Multi-Model Particle Filtering (SDCMMPF) method, in which the inherently vague human-generated data are modeled using a Fuzzy Inference System (FIS). The soft data are then transformed into a set of constraints, which enable the MMPF method to deal with tracking situations involving potentially highly agile targets. The experimental results demonstrate the capability of the proposed SDCMMPF to significantly outperform the conventional MMPF when applied to various agile target-tracking scenarios.

I. INTRODUCTION

The problem of tracking targets whose dynamics include multiple-switching regimes, also known as maneuvering target tracking, has been studied extensively as reflected in the review paper series by Li & Jilkov [1], [2], [3]. The maneuvering target tracking problem in the presence of non-linearity has also been studied, and several methods such as function approximation, sampling-based moment approximation, and stochastic model approximation have been proposed to tackle this issue [4]. The sampling-based moment approximation methods using the sequential Monte Carlo approach, also known as particle filtering, are widely deployed to deal with non-linear tracking problems. An extension of the particle filtering methodology according to the multi-model principles, i.e. MMPF, provides a powerful and flexible solution to non-linear maneuvering target tracking problems. The uncertainty regarding the target mode and its transitions are typically characterized in a Markovian manner, using the so-called transition probability matrix [5].

In this paper, we consider the problem where dynamics of the maneuvering target might deviate from the probabilistic characterization represented by the transition probability matrix. We refer to this problem as agile target tracking, and consider agility levels to be directly associated with the likelihood of unpredictable target maneuvers. If a target is agile, its next mode, which is assumed to be one of the available models, is unpredictable. Agile target tracking is an important and challenging problem, which, to the best of our knowledge, has rarely been addressed in the literature. This lack of interest is partly due to the difficulty of obtaining data regarding the agility level of targets using the conventional sensory mechanisms. On the other hand, a relatively recent trend in the data fusion community aims at exploiting data provided by humans [6], [7], [8]. Our observation is that human agents have advanced cognitive abilities, which allow them to provide valuable information regarding intricate target behaviors, including the agility. Accordingly, the proposed approach in this paper deploys human-generated data regarding a target’s agility level to improve the performance of an MMPF target-tracking method.

A popular approach to enhance the performance of Bayesian filtering methods using extra information and subsequent incorporation of specific constraints into the filtering process is introduced in [9]. Following the same principle, we propose a soft-data-constrained MMPF method; where the inherently vague (subjective) soft data provided by human agents are modeled using a Fuzzy inference system. They are then transformed into a set of constraints, which are enforced to enable dealing with tracking situations involving potentially highly agile targets. The extensive experiments conducted for the task of single agile target tracking demonstrated the efficiency of the proposed approach in enhancing the capacity of the MMPF method. In particular, the conventional MMPF method is shown to diverge when applied to highly agile targets. In comparison, the proposed soft-data-constrained MMPF is capable of tracking highly agile targets when provided with appropriate soft data.

The rest of this paper is organized as follows. An overview of the related literature work is presented in section 2. The proposed soft-data-constrained MMPF is detailed in section 3. The experimental results obtained for the task of single agile target tracking are provided in section 4. Finally, the concluding remarks along with a discussion of potential areas of future work are presented in section 5.

II. RELATED LITERATURE

A. Constrained Bayesian Filtering

Attempts to exploit the external knowledge as constraints to improve tracking performance can be traced back to the early 1990s [10]. The literature related to constrained Bayesian filtering contains a wide spectrum of techniques, including pseudo-measurement [11], clipping [12], projection [13], and optimization-based methodologies [14].

The formalized constrains themselves can be of various types and forms, such as linear, non-linear, soft, hard, equal-
ity, and inequality [15]. Constrained variants of the particle filtering method have also been proposed in the literature, assuming a variety of domain-specific constraints [16], [38]. As discussed by Simon [9], for the case of linear systems with linear constraints, all of the existing approaches result in the same optimal state estimate. On the other hand, for non-linear cases, the number of state estimation techniques can be overwhelming, as the constrained filtering problem can be viewed from many different perspectives. Consequently, research on the theory and implementation of constrained particle filters remains an active area, with much room for future work.

B. Soft Data Fusion

A recent trend in the data fusion community aims at exploiting data provided by humans. In contrast to the conventional data provided by well-calibrated sensors, also referred to as hard data, human-generated data, known as soft data [6], [17], are typically unstructured, vague, and subjective. On the other hand, the main advantage of humans is their ability to perform complex cognitive tasks. They can provide high level data regarding the targets that could be very difficult, if not impossible, to obtain using hard conventional sensors. A tremendous amount of research has been done on data fusion using conventional sensors. In contrast, limited work has studied the fusion of data produced by human and non-human sensors. Hall et al. [31] provide a brief review of ongoing work on dynamic fusion of hard/soft data, identifying its motivation and advantages, challenges, and requirements. A recent preliminary research in this area is the work on generating a dataset for hard/soft data fusion intended to serve as a foundation and a verification/validation resource for future research [32]. Very recently, a Dempster-Shafer theoretic framework for soft/hard data fusion was proposed that relies on a novel conditional approach to updating, as well as a new model to convert propositional logic statements from text into forms usable by Dempster-Shafer theory [33].

Another trend of work along this area is focused on the so called human centered data fusion paradigm and puts emphasis on the human role in data fusion process [35], [34]. This new paradigm considers humans as active participants in the data fusion process and not merely as soft sensors but also as hybrid computers and ad-hoc teams (hive mind). It relies on emerging technologies such as virtual worlds and social network software to support humans in their new fusion roles. In spite of these accomplishments, research on hard/soft data fusion, as well as human-centered fusion is still in its fledging stage and should provide rich opportunities for further theoretical advancement and practical demonstrations in the future [17].

C. Multi-Model Maneuvering Target Tracking

Among the plethora of methodologies developed in the literature on maneuvering target tracking, the multi-model techniques are probably the most popular [3]. The multi-model tracking methods belong to the family of hybrid estimation techniques in which, the target state includes both continuous and discrete components. The base target state is the component that varies continuously as in conventional tracking systems. The discrete component, however, has a stair-case type trajectory, i.e., it may either jump or remain unchanged, which is commonly known as the target mode. The conventional solution to multi-model tracking is to follow the estimation after the decision approach, i.e., first deciding on the best mode of the target and then applying a single filtering process using the chosen mode as if it is the correct one [3].

In [18] the multi-model tracking algorithms have been categorized into three generations, in which each new generation is deemed to be fundamentally different in terms of operation, structure, and capabilities. The first generation, typically referred to as the Autonomous Multi-Model (AMM) filtering, has the distinctive characteristic that each one of the elemental filters operates independently [19]. The second generation, Interacting Multiple Models (IMM), improves upon the first by enabling the elemental filters to operate in a more efficient cooperative manner through effective interactions [20]. Finally, the third and the most recent generation adds the benefits of variable structure filtering, e.g. constant adaptation of mode transition probabilities to further enhance the performance [21]. Accordingly, the constrained MMPF method proposed in this paper can be considered as a variant of the variable structure MMPF algorithms.

III. SOFT-DATA-CONSTRAINED MMPF

This section represents the proposed soft-data-constrained MMPF method, first discussing a generic MMPF [5], used as the underpinning algorithm of the proposed constrained filtering method. Subsequently, the procedure used to model and incorporate soft human-generated constraints into the generic MMPF method is detailed.

A. Multi-Model Particle Filtering

Multi-model particle filtering has been proposed by several authors [23], [24], [25], to perform nonlinear filtering with switching dynamic modes. Algorithm 1 shows the pseudocode for a generic MMPF [5]. The MMPF operates as a general discrete-time hybrid system, which is modeled by the following target dynamic and measurement models, respectively:

\[
x_t = f_{t-1}(x_{t-1}, r_t) + \varepsilon_{t-1}(r_t)
\]

\[
z_t = h_t(x_t, r_t) + \delta_t
\]

where the covariance matrices of process noise, \( \varepsilon_{t-1} \), and measurement noise, \( \delta_t \), are \( Q_{t-1}(r_t) \) and \( R_t \), respectively. The \( r_k \in S = \{1, \ldots, s\} \) is the regime (mode) variable in effect during the sampling process, and the target state is represented as an augmented hybrid-state vector defined as \( y_t = [x_t^T r_t]^T \).

This algorithm input includes \( y^*_n \), which consists of both the state \( x^n_t \) and the mode \( r^n_t \) of each particle. In these terms, \( n = 1, \ldots, N \) and \( N \) is the total number of particles. The other inputs are the particles’ normalized weights, \( w^n_t \), and
Algorithm 1: Generic MMPF

\[
[y^n_t, w^n_t]_{n=1}^N = \text{MMPF} \left[ [y^n_{t-1}, w^n_{t-1}]_{n=1}^N, z_t \right]
\]

Step 1: Regime transition (RT):

\[
\{r^n_t\}_{n=1}^N = \text{RT} \left[ \{r^n_{t-1}\}_{n=1}^N, \Pi \right]
\]

Step 2: Regime conditioned SIS:

\[
\{y^n_t, w^n_t\}_{n=1}^N = \text{RC-SIS} \left[ \{x^n_{t-1}, r^n_{t-1}\}_{n=1}^N, \hat{z}_t \right]
\]

Step 3: \( N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N} (w^n_t)^2} \)

Step 4: If \( N_{\text{eff}} < N_{\text{thr}} \) Resample:

\[
\{y^n_t, w^n_t\}_{n=1}^N = \text{Resample} \left[ \{y^n_t, w^n_t\}_{n=1}^N \right]
\]

End If

the observation at time \( t \), denoted by \( z_t \). The algorithm output is the particles’ new states, modes, and weights.

**Step 1** aims to predict the next set of particles’ modes, \( \{r^n_t\}_{n=1}^N \), based on their previous modes, \( \{r^n_{t-1}\}_{n=1}^N \), and the transition probability matrix \( \Pi = [\pi_{ij}] \), where \( i, j \in S \). If \( r^n_{t-1} = i \), then \( r^n_t \) should be set to \( j \) with a probability equal to \( \pi_{ij} \); therefore, if \( r^n_{t-1} = i \) and \( u_n \tilde{U}(0,1) \), then \( r^n_t \) is set to \( m \in S \) such that:

\[
\sum_{j=1}^{m-1} \pi_{ij} < u_n \leq \sum_{j=1}^{m} \pi_{ij}
\]

The cumulative distribution function of discrete random variable \( r_t \) given \( r_{t-1} = i \) is given by \( \sum_{j=1}^{m} \pi_{ij} \). **Step 2** is Sequential Importance Sampling (SIS), which involves both prediction and update sub-steps as described in the following. In the prediction step, the next state is predicted based on the previous state \( x_{t-1} \) and observations up to time \( t-1 \), denoted as \( z_{1:t-1} \).

\[
p(x_t, r_t = j | z_{1:t-1}) = \sum_{i} \pi_{ij} \int p(x_t | x_{t-1}, r_t = j) \times p(x_{t-1}, r_{t-1} = i | z_{1:t-1}) dx_{t-1}
\]

(4)

During updating, this prediction gets updated based on the current measurement at time \( t \), that is, \( z_t \).

\[
p(x_t, r_t = j | z_t) = \frac{p(z_t | x_t, r_t = j) p(x_t, r_t = j | z_{1:t-1})}{\sum_{i} \int p(z_t | x_t, r_t = j) p(x_t, r_t = j | z_{1:t-1}) dx_t}
\]

(5)

A suitable measure of an algorithm’s degeneracy is the effective sample size, \( N_{\text{eff}} \), which is calculated in **step 3** of the algorithm. In **step 4**, \( N_{\text{thr}} \) is used as a threshold such that, if \( N_{\text{eff}} \) falls below that value, then the resampling step is required.

**B. Soft Data Modeling**

To model the soft data report, which is supplied by a human observer, the reports are assumed to comply by a specific syntax and semantics, which are predefined by an ontology. Please note that an appropriate Natural Language Processing (NLP) method can be used to format raw soft data according to this syntax. The syntax for the soft data report is shown in Figure 1. As shown in this figure, each report is a natural language expression that reports on the agility level of the target along with the level of certainty presumed by the reporter. In other words, each report is an expression comprised of a target-identification term, target ID, a qualifier term to express the level of certainty presented by the report, and a term to represent the perceived agility level of the target.

**Fig. 1. The syntax considered for the soft data reports**

Fuzzy inference systems are used to capture the uncertainty arising from the vagueness of soft data. One could argue in favor of probabilistic approaches as an alternative to fuzzy inference for soft data processing. However, a closer examination reveals that probabilistic measures are appropriate when dealing with ill-defined (random) variables hitting well-defined sets, whereas fuzzy measures enable calculating the membership of well-known variables in ill-defined (vague) sets [39]. The semantics used to interpret the given soft data is as follows, with three different categories of Reported Agility Level (RAL) and Reported Certainty Level (RCL).

For the RAL, the report can be judged extremely, highly, or marginally agile and for the RCL, it can be considered certainly, almost, or perhaps certain. As a result, we have modeled nine different FISs that have different rules and different membership functions for the output variable. Based on the agility level reported, RAL, and the certainty level of the report, RCL, one of the fuzzy models is chosen. A simple decision tree is deployed to accomplish the fuzzy model selection. Based on the inputs RCL={“slightly”, “perhaps”, “certainly”) and RAL={“extremely”, “highly”, “marginally”) which are mapped respectively to be \( \{0, 1, 2\} \), one of the models is chosen. For example if the input is “robot is
certainly extremely agile”, i.e., \( RCL = 2 \) and \( RAL = 0 \), by navigating through the tree, the appropriate model is selected. After choosing the FIS, the Expected Cluster Weight (ECW), which represents the MMPF measure, along with the Stochastic Agility Discount (SAD), which is the respective values of the transition matrix, are the inputs to the fuzzy system that outputs the constraints \( C_{m,m'} \).

Table I shows the rules defined when \( RAL \) is “extremely”, and Table II depicts a case in which the \( RAL \) is “marginally/not”. The rules of the FIS change based on different values reported for \( RAL \), therefore we have nine different sets of fuzzy rules. Table I and Table II demonstrate the rules defined for two different \( RALs \). In these tables, the terms “vlow”, “med” and “vhigh” represent very low, medium and very high respectively. The value reported for RCL affects the membership function of the output values \( C_{m,m'} \). As the certainty level increases, the membership functions get more crisp, i.e., narrower. The following section describes the procedure of incorporating soft data as dynamic constraints into MMPF.

C. Incorporating Soft Data As Dynamic Constraints

Algorithm 2 depicts the pseudocode of the proposed Soft-Data-Constrained Multi-Model Particle Filter (SDCMMPF).

The algorithm starts with \textit{step 1}, which consists of the uniform initialization of the particle clouds; that is, each of the possible target modes is represented by the same number of particles, \( \frac{N}{M} \). Each particle having the same weight computed as \( \frac{1}{N} \), in which, \( N \) and \( M \) are the total number of the particles and the number of target modes, respectively. Our contributions are focused in \textit{step 2}, the mode prediction step. The generic MMPF prediction aims at simulating the target mode transition probabilities dictated by matrix II. Due to the stochastic nature of this process, the predictions of MMPF are always slightly different from those that result from II.

For agile targets, the higher the agility level, the less the likelihood of the next target mode being the same as the mode predicted by II. Accordingly, our main objective is to reinforce the aforementioned stochastic difference, if the target is reported to be agile and vice versa; that is, discouraging this difference for targets with no/marginal agility. To achieve this goal, in \textit{step 2.1}, particles are clustered based on their current mode. The next target mode, \( r_{t} \), is predicted using the previous mode, \( r_{t-1} \), along with the transition matrix, II in \textit{step 2.2.a}. At the same time, next target mode, \( r_{t}^{*} \), is predicted using \( r_{t-1} \) along with the regular MMPF in \textit{step 2.2.b}. Next, the difference between these two particle clouds, i.e., \( r_{t} \) and \( r_{t}^{*} \), is calculated using KLD measure in \textit{step 2.2.c}, followed by normalizing this value in \textit{step 2.2.d}.

The KLD measure between the two distributions which are predicted by transition probability matrix and the MMPF is calculated as follows [26]. Consider two distributions \( p \) and \( q \) and let the two sets \( \{X_1, ..., X_n\} \) and \( \{Y_1, ..., Y_m\} \) be i.i.d samples drawn independently from the \( p \) and \( q \), respectively. In [27], an asymptotically unbiased and mean-square consistent estimator \( \hat{D}_{KL}(p,q) \) of \( D_{KL}(p,q) \), based on the k-Nearest Neighbor (NN) density estimation [28], is defined as in the following:

\[
\hat{D}_{KL}(p,q) = \frac{d}{n} \sum_{i=1}^{n} \log \frac{\nu_{k_i}(i)}{\rho_{k_i}(i)} + \frac{1}{n} \sum_{i=1}^{n} \psi(l_i) - \psi(k_i) + \log \frac{m}{n-1}
\]  

(6)

where \( \nu_{k_i}(i) \) is the Euclidian distance from \( X_i \) to its \( k_i \)-NN in \( \{Y_j\} \), the \( \rho_{k_i}(i) \) is the Euclidian distance between \( X_i \) and its \( l_i \)-NN in \( \{X_j\}_{j \neq i} \). The \( \psi \) is the Digamma function, defined as the logarithmic derivative of the Gamma function. Figure 2 shows the steps in which KLD measures are calculated for all particle clouds corresponding to each of the target modes.

In \textit{step 2.3}, soft data is modeled using a fuzzy inference system as discussed in the previous sub-section. One of the FISs is selected based on the RCL and RAL of the human report in \textit{step 2.3.a}, and then ECW along with the SAD are the input to the selected FIS, shown in \textit{step 2.3.b}. As shown in \textit{step 2.4}, each of the particle clouds \( PC_{m}, m = 1, ..., M \) are further
clustered into \( M \) sub-clusters at step 2.4.a. The constraint weights \( C_{m,m'} \) are calculated in step 2.4.b as follows: If the target agility level, which is input by the user, is high and the estimation of the target location is close to that predicted by the transition probability matrix, i.e., a small KLD measure, then the particles that follow the behavior defined by \( \Pi \) should get low weights and gradually disappear. On the other hand, the rest of the particles, which are not behaving similar to model \( \Pi \), should get higher weights, in order to duplicate (survive) more in the resampling step. The soft-data-inspired dynamic constraints affect the particles’ weights before the resampling step, i.e., the weights are imposed in PF onto prior particles; therefore, the weighting of the particles is as follows:

\[
    w_{t}^{n} = w_{t-1}^{n} p(z|x) C_{m,m'}
\]

in which the constant weights \( C_{m,m'} \), defined as \( C_{m,m'} = p(SD|x) \), are calculated in step 2.4.i of the algorithm. The following section presents the results of the experiments conducted to compare the performance of the proposed method and the generic MMPF.

IV. SINGLE AGILE TARGET TRACKING EXPERIMENTS

Three categories of experiments were conducted. In the first category, the effect of human agent’s reports on different target agility levels was evaluated. In the second category, the aim was to measure the impact of the level of uncertainty in these human agent’s reports. The third-category experiments presented were conducted to show the robustness of the proposed method to the varying agility levels.

A. Experimental Setting

The baseline used to assess the experimental results achieved by the proposed method is the generic MMPF, and the metric for evaluating the performance is the Mean Squared Error (MSE) between the estimated and the original target trajectories. In the experiments, the transition probability matrix \( \Pi \) was used which is defined as follows:

\[
    \Pi = \begin{bmatrix}
        0.05 & 0.15 & 0.8 \\
        0.8 & 0.1 & 0.1 \\
        0.1 & 0.8 & 0.1 
    \end{bmatrix}
\]

The following modes indicate the behavior of a target: mode 1, moving straight East, mode 2, moving straight South East (SE) and finally, mode 3 moving North East (NE). For instance, to transit from mode 3 to mode 2, the target turns \(-90^\circ\) and continues moving straight. Based on the transition matrix, when a target is at mode 1, it will most probably next transit to mode 3, and so on. Each target has periodic behavior with three maneuvers per period. In order to simulate medium agility, 1 or 2 (out of 3) maneuvers do not take place according to \( \Pi \). When there is a high level of agility, no maneuvers take place according to \( \Pi \). For example, as shown in Figures 5.a and 7.a, the target is at mode 1 and remains at that mode.

B. Category I: Impact of target agility level

In this category of experiments, three different scenarios were designed and performed and reported on by a human agent. In scenario 1.a, shown in Figure 3.a, the report was “robot is certainly marginally/not agile.” In scenario 1.b,
Figure 3.a: True & estimated target trajectory for the scenario I.a

Figure 3.b: Performance comparison: low agile target & highly certain SD

Figure 4.a: True & estimated target trajectory for the scenario I.b

Figure 4.b: Performance comparison: agile target & highly certain SD

Figure 5.a: True & estimated target trajectory for the scenario I.c

Figure 5.b: Performance comparison: highly agile target & highly certain SD

Figure 6.a: True & estimated target trajectory for the scenario II.a

Figure 6.b: Performance comparison: agile target & uncertain SD
shown in Figure 4.a, the report was “robot is certainly highly agile,” and the target was tasked to periodically maneuver as follows:

\[ \text{mode} 1 \rightarrow \text{mode} 2 \]

\[ \text{mode} 2 \rightarrow \text{mode} 1 \]

Finally, in scenario I.c shown is Figure 5.a, the report was “robot is certainly extremely agile” and the target was tasked to maneuver as follows:

\[ \text{mode} 1 \rightarrow \text{mode} 1 \]

As shown in Figures 3.b and 4.b, the proposed method improves tracking performance for both medium and no agility. In the case of an extremely agile target, as shown in Figure 5.a, regular MMPF cannot track the target. However, using soft data, the proposed method assigns higher weights to the particles which do not follow the maneuver characteristics defined by matrix II, and assigns lower weights to the rest. The particles assigned higher weights will obtain even higher weights in the updating step, as they represent target state more agreeable with the measurements. Consequently, in the resampling step, they survive and regenerate more; therefore, the estimation improves after some iterations.

C. Category II: Impact of SD certainty level

In these set of experiments, the aim was to assess the effect of a human agent’s level of uncertainty on estimation accuracy. Figures 6.a to 7.b show the results in which the human agent’s uncertainty regarding the report was high. In the first scenario, the report was “robot is perhaps highly agile,” and in the second one, the report was “robot is perhaps extremely agile.” As shown in Figures 6.a to 7.b, when the certainty of the report given by the human agent was low; the estimation was not as accurate as the results obtained in the previous category, in which the user reported the agility level of the target with higher certainty.

D. Category III: Impact of varying target agility level

In this experiment, the aim was to examine the robustness of the proposed method to varying agility levels, which is the most realistic case in practice. In this case, the target first maneuvers based on the characteristics defined by matrix II, and after a time, it may start to deviate from that. Thus, the level of agility increases as time goes by. In this scenario, the report was given according to the situation of the target observed by the agent. The target was maneuvering without agility at the beginning, and after some iterations, it moved with a medium level of agility; it then continued with high level of agility. Figure 8.a shows the true and the estimated target trajectories based on MMPF and the proposed method. As shown, the proposed method can track a target while it is switching its behavior from not agile to highly agile; however, MMPF fails to track the target precisely and gets lost when the agility level becomes high. Figure 8.b shows the performance comparison of the last scenario.

V. CONCLUSION

This paper has proposed a Soft-Data-Constrained Multi-Model Particle Filtering (SDCMMPF) method, in which inherently vague soft data provided by human agents are properly modeled using a Fuzzy inference system. These data are then transformed into a set of constraints and imposed on the
MMFP method, enabling it to deal with tracking situations involving potentially highly agile targets. The experiments conducted for the task of single agile target tracking demonstrate the proposed approach’s efficiency in enhancing the MMFP method’s ability to deal with target agility. The method meets this goal by incorporating the agility level reported as soft data into the tracking process as dynamic constraints. In particular, the conventional MMFP method is shown to perform poorly, that is, it diverges when applied to highly agile targets. However, the proposed method is capable of tracking highly agile targets when provided with appropriate soft data.

As future work, we plan to extend the proposed soft-data-constrained MMFP method to enable distributed constrained tracking of agile targets, assuming the application platform to be a sensor network comprised of a large number of trackers. This method would allow large scale deployment of the proposed approach to real-world tracking problems, while potentially further improving the performance by exploiting extra information provided through a large number of human agents.

REFERENCES