Distributed Target Tracking for Nonlinear Systems: Application to Bearings-Only Tracking

Thomas Bréhard
IRISA/CNRS
IRISA, Campus de Beaulieu
35042 Rennes Cedex
France
Email: tbrehard@irisa.fr

Jean-Pierre Le Cadre
IRISA/CNRS
IRISA, Campus de Beaulieu
35042 Rennes Cedex
France
Email: lecadre@irisa.fr

***Abstract*— Nonlinear distributed target tracking for a single target is addressed in this paper. The problem consists in deriving fusion rules for local full/partial target state estimates processed by a number of sensors. We investigate the general ways for the nonlinear fusion rules with/without feedback implementation via particle filtering algorithms. In particular, we focuse on practical application of these ideas for specific multi-sensor architectures including low/high bandwidth. Then, these new approaches are applied to the distributed bearings-only tracking problem.**

NOTATION

$\|X\|_Q^2$: $\|X\|_Q^2 = X^* Q^{-1} X$ where X is a column vector,

PDF : Probability Density Function,

X^* : denotes the transpose of matrix X.

I. INTRODUCTION

In many applications, multiple sensors are used to collect noise corrupted observations about target's trajectory. This problem of tracking has been of interest for the past thirty years [1]. A centralized architecture which consists in transmitting measurements directly to a fusion node is theoretically optimal. However, this approach has some disadvantages. First, this architecture needs a high bandwidth to collect measurements. Moreover, the fusion node needs high computation abilities. Consequently, along the past fifteen years, the distributed approach has received a lot of interest. In this approach, each observer processes its own observation history to obtain a local conditional Probability Density Function (PDF) of the target state. Then, each local conditional PDF is sent to the fusion node. The fusion estimation problem consists in constructing the global conditional PDF of the target's trajectory.

The multisensor tracking problem is composed of $S + 1$ stochastic equations. The first one represents the temporal evolution of the target state (position and velocity), called state equation. The S other ones link the measurement obtained by the S sensors to the target state at time t (measurement equations).

The problem of the distributed tracking composed of linear equations with Gaussian noises has been widely addressed in the literature. In this case, each sensor transmits a sufficient statistic which is provided by the local conditional mean and covariance via a Kalman filter. Moreover it has been shown that the global conditional mean and covariance can be obtained via linear operations on local estimates. This is the BLUE fusion rule proposed in [2]. In 1990, Chong et al. in [3] proposed a distributed Kalman filtering fusion with feedback. In the latter, the fusion node sends its latest estimate to local sensors. In 2001, Zhu et al. showed in [4] that a feedback step does not improve tracking performance at the fusion node level but reduces the covariance of each local estimate.

In the nonlinear case, few results have been obtained yet. At the sensor level, observer maintains its own estimate using a nonlinear tracking algorithm like a particle filter. This algorithm based on importance sampling and Monte-Carlo methods, is quite convenient in nonlinear/non-Gaussian filtering problems. Arulampalam et al. in [5] gave a good overview of particle filtering algorithms. At the fusion level, Nguyen et al. [6] investigated particle filter as a fusion rule in a distributed tracking problem for a maneuvering target. They proposed to consider local estimates as measurements which are fused using a particle filter. However, the authors did not used the nonlinear fusion equations given by Chong et al. in [3] and Castanon et al in [7]. As a matter of fact, these equations do not have analytical formulas such that particle filter is quite convenient as a fusion rule. We derive here distributed tracking algorithms with and without a feedback step based on the nonlinear fusion equations via particle filtering algorithms.

In this paper, an example of distributed target tracking using bearings-only measurements is presented. This problem has been studied by Chong et al [3] in 1990 and Anderson and Itlis [8] in 1993. In the latter, the authors proposed a local tracking algorithm based on reduced sufficient statistics. This algorithm is compared with a distributed extended Kalman filter using the modified polar coordinates system. The underlying idea is that local conditional mean and covariance are not sufficient statistics which could be broadcasted to the fusion node in the bearings-only context. We propose in this paper a local tracking algorithm based on a particle filter. This algorithm is quite convenient in bearings-only tracking applications according to Ristic et al. in [9].

However, the main characteristic of the bearings-only tracking problem is the weak observability of range at sensor level [10] such that tracking algorithms are rather difficult to initialize [11] and sometimes divergent. Then, we propose to process only a partial target state at the sensor level with a robust algorithm. Only partial state estimates are broadcasted to the fusion node. When fusing partial state estimates, range is estimated such that the full target state is computed. We derive, in this paper, distributed tracking algorithms with and without feedback based on the general fusion equations adapted to partial local state estimates.

From a practical point of view, sensor architectures include high or low bandwidth features. If a high bandwidth feature is available, then each sensor broadcasts the local conditional PDF given by the particle filtering algorithm. In this case, the transmitted information is a sum of weighted Dirac delta functions. If sensor architecture includes only low bandwidth feature then each sensor transmits only local conditional mean and covariance estimates. We precise in this paper the nonlinear fusion equations in the context of bearings-only tracking when sensors architecture includes a high/low bandwidth feature.

The distributed target tracking problem is presented in section II. We focus on the information transmitted by each sensor whether sensor architectures include a feedback step, or not. In section III, we consider that each sensor broadcasts a local conditional estimate of the full target state. Distributed target tracking algorithms based on the general nonlinear fusion equations via particle filter are derived. In section IV, we investigate the case where each sensor broadcasts a local conditional estimate of the partial target state. Distributed target tracking algorithms are derived. The approach presented in section III and IV are applied to the distributed bearings-only tracking in section V. In particular, the nonlinear fusion equations are precised in the case of low and high bandwidth in this context. Simulation results illustrate the different sensors architectures in section VI.

II. Problem

S sensors are used to collect noise corrupted observations about target's trajectory. First, let us define:

- X_t the target state at time t composed of target's relative velocity and position,
- $X_{0:t}$ the trajectory of the target from time 0 to time t,
- Z_t^s the measurement received at time t by sensor s,
- $Z_{1:t}^s$ the measurements obtained by sensor s, from time 1 to time t,
- $Z_{1:t}^{1:S}$ the set of measurements obtained, from time 1 to time t, by **all** the S sensors.

The target state is related to the measurements through the following equation:

$$p(Z_t^s|X_t) \ , s \in \{1, \dots, S\} \ . \tag{1}$$

Moreover, we assume a prior information on target's diffusion:

$$p(X_{t+1}|X_t) \ . \tag{2}$$

The aim of the distributed target tracking problem is to estimate, for a known function f:

$$\mathbb{E}\{f(X_{0:t})|Z_{1:t}^{1:S}\} \ . \tag{3}$$

To compute the term given by eq.(3), the fusion node can use the prior information on target i.e. $p(X_{t+1}|X_t)$ and the information broadcasted by local sensors. These sensors process their local conditional estimates by applying a tracking algorithm with eq.(1) as measurement equation, and eq.(2) as prior information on target's diffusion. However, transmitted information is different if architecture includes or not a feedback step. As a matter of fact, if there is no feedback step, each sensor maintains its estimate using its own observation history such that at time t, sensor s sends $\hat{p}(X_t|Z_{1:t}^s)$ i.e. an *estimate* of $p(X_t|Z_{1:t}^s)$. If there is a feedback step, then the fusion node broadcasts its latest estimates i.e. $\hat{p}(X_{t-1}|Z_{1:t-1}^{1:S})$ to local sensors at time t. Consequently, sensor s uses this PDF and its current measurement Z_t^s to obtain $\hat{p}(X_t|Z_{1:t-1}^{1:S}, Z_t^s)$ i.e. an estimate of $p(X_t|Z_{1:t-1}^{1:S}, Z_t^s)$. These ideas, summed up in tab.I, will be helpful in section III to understand how nonlinear fusion rules based on particle filtering algorithms can be derived.

Architecture	**Sensor**
Without feedback	$\hat{p}(X_t|Z_{1:t}^s)$
With feedback	$\hat{p}(X_t|Z_{1:t-1}^{1:S}, Z_t^s)$

TABLE I

Local information sent by sensor s to the fusion node.

However, in some applications, sensors can only broadcast partial information. For example, in the bearings-only context, range at the sensor level is not always observable (see [10]) such that each sensor transmits only a partial state estimate. We note Y_t^s the partial state for sensor s at time t. This conditional partial state estimate is processed using a tracking algorithm including both $p(Z_t^s|Y_t^s)$ as measurement equation and $p(Y_{t+1}^s|Y_t^s)$ as prior information on partial state. The estimation of the full state is processed only at the fusion node level. We assume that there is a known function h such that we can compute the full state X_t from the set of partial states $Y_t^{1:S}$. Moreover, we assume the existence of S known functions noted $\{g_s\}_{s\in\{1,\dots,S\}}$ such that the partial target state of sensor s can be computed from the full state X_t.

$$\begin{cases} X_t &= h(Y_t^{1:S}) \ , \\ Y_t^s &= g_s(X_t) \ , \quad \forall s \in \{1, \dots, S\} \ . \end{cases} \tag{4}$$

In this case, the fusion node computes eq.(3) using the prior information on the target i.e. $p(X_{t+1}|X_t)$ and the partial information broadcasted by local sensors. This information is different whether or not architecture includes a feedback step. The PDF transmitted by sensor s to the fusion node at time t for the different cases is summed up in tab.II. The nonlinear fusion rules based on particle filtering algorithms are derived in section IV.

Architecture	Sensor
Without feedback	$\hat{p}(Y_t^s\|Z_{1:t}^s)$
With feedback	$\hat{p}(Y_t^s\|Z_{1:t-1}^{1:S}, Z_t^s)$

TABLE II

Local partial information sent by sensor s to the fusion node.

III. NONLINEAR FUSION RULES FOR LOCAL FULL ESTIMATES

The aim of this section consists in deriving fusion rules in the context of nonlinear distributed tracking with and without feedback when each sensor broadcasts a local estimate of X_t at time t. These fusion rules must only depend on the prior information about the target i.e. $p(X_{t+1}|X_t)$ and the information broadcasted by local sensors (see tab.I). First, let us remind that the aim is to estimate, for a known function f:

$$\mathbb{E}\{f(X_{0:t})|Z_{1:t}^{1:S}\} = \int f(X_{0:t})p(X_{0:t}|Z_{1:t}^{0:S})dX_{0:t} \; . \qquad (5)$$

We can see that eq.(5) is a complex integral which does not have a closed-form expression. An idea consists in using particle filtering algorithm. We use the same approach as in [12] to solve the problem. First, an importance function $\pi(X_{0:t}|Z_{1:t}^{1:S})$ is introduced such that eq.(5) can be rewritten:

$$\mathbb{E}\{f(X_{0:t})|Z_{1:t}^{1:S}\} = \int f(X_{0:t})w_t\pi(X_{0:t}|Z_{1:t}^{1:S})dX_{0:t}$$

where

$$w_t = \frac{p(X_{0:t}|Z_{1:t}^{1:S})}{\pi(X_{0:t}|Z_{1:t}^{1:S})} \; . \qquad (6)$$

The classical approach consists in sampling N trajectories, denoted $\{X_{0:t}^i\}_{i=1,\dots,N}$, using the importance function and computing the weights $\{w_t^i\}_{i=1,\dots,N}$ such that:

$$\mathbb{E}\{f(X_{0:t})|Z_{1:t}^{1:S}\} \approx \sum_{i=1}^{N} f(X_{0:t}^i)w_t^i \; . \qquad (7)$$

The problem is that the $\{w_t^i\}_{i=1,\dots,N}$ cannot be computed because $p(X_{0:t}|Z_{1:t}^{1:S})$ is unknown in eq.(6). We demonstrate in the two following sections that $\{w_t^i\}_{i=1,\dots,N}$ can be computed recursively using the diffusion equation $p(X_{t+1}|X_t)$ and the local conditional PDFs processed by the S sensors. Moreover, we precise how trajectories $\{X_{0:t}^{s,i}\}_{i=1,\dots,N}$ can be sampled using the importance function. In the first section, sensors architecture does not include feedback feature, while sensors architecture in the second one does.

A. Without feedback

We address in this section the particle filtering algorithm used as a nonlinear fusion rule when sensors architecture does not include a feedback step. A recursive formula for $\{w_t\}_{i=1,\dots,N}$ is derived and only depends on the information available at the fusion level. First, we show in appendix A that a recursive formula for $p(X_{0:t}|Z_{1:t}^{1:S})$ is given by:

$$p(X_{0:t}|Z_{1:t}^{1:S}) \propto \prod_{s=1}^{S}\left\{\frac{p(X_t|Z_{1:t}^s)}{p(X_t|Z_{1:t-1}^s)}\right\} \times p(X_t|X_{t-1})p(X_{0:t-1}|Z_{1:t-1}^{1:S}) \; . \qquad (8)$$

Let us notice that eq.(8) is the classical nonlinear fusion equation when there is no feedback as given by Chong et al. in [3]. We classically assume that the importance function is such that:

$$\pi(X_{0:t}|Z_{1:t}^{1:S}) = \pi(X_t|X_{0:t-1}, Z_{1:t}^{1:S})\pi(X_{0:t-1}|Z_{1:t-1}^{1:S}) \; . \qquad (9)$$

Then, incorporating eq.(8) and eq.(9) in eq.(6), we obtain:

$$w_t \propto \prod_{s=1}^{S}\left\{\frac{p(X_t|Z_{1:t}^s)}{p(X_t|Z_{1:t-1}^s)}\right\} \times \frac{p(X_t|X_{t-1})}{\pi(X_t|X_{0:t-1}, Z_{1:t}^{1:S})}w_{t-1} \; . \qquad (10)$$

We use the prior information on target $p(X_t|X_{t-1})$ as importance function:

$$\pi(X_t|X_{0:t-1}, Z_{1:t}^{1:S}) = p(X_t|X_{t-1}) \; . \qquad (11)$$

Consequently, trajectories $\{X_{0:t}^i\}_{i=1,\dots,N}$ are sampled using the prior information on target's trajectory. Of course, more accurate strategies can be proposed, like studying the optimal importance sampling proposed by Doucet in [13] or using auxiliary particle filter in [14]. Incorporating eq.(11), eq.(10) becomes:

$$w_t \propto \prod_{s=1}^{S}\left\{\frac{p(X_t|Z_{1:t}^s)}{\int p(X_t|X_{t-1})p(X_{t-1}|Z_{1:t-1}^s)dX_{t-1}}\right\} w_{t-1} \; . \qquad (12)$$

Finally, using the plug-in method which consists in replacing an unknown PDF by an estimated PDF, eq.(12) becomes:

$$\boxed{w_t \propto \prod_{s=1}^{S}\left\{\frac{\hat{p}(X_t|Z_{1:t}^s)}{\int p(X_t|X_{t-1})\hat{p}(X_{t-1}|Z_{1:t-1}^s)dX_{t-1}}\right\} w_{t-1}} \; . \qquad (13)$$

Consequently, the particle filtering algorithm is composed of:

- a recursive formula for w_t given by eq.(13) which depends on $\{\hat{p}(X_t|Z_{1:t}^s)\}_{s=1,\dots,S}$, $\{\hat{p}(X_{t-1}|Z_{1:t-1}^s)\}_{s=1,\dots,S}$ (i.e. the information broadcasted by sensor s at time t and $t-1$) and $p(X_t|X_{t-1})$ (i.e. the prior information on target's trajectory).
- trajectories $\{X_{0:t}^i\}_{i=1,\dots,N}$ sampled using $p(X_t|X_{t-1})$.

The distributed tracking algorithm is summed up in fig.1.

B. With feedback

We derive in this section the particle filtering algorithm used as a fusion rule when sensors architecture includes a feedback step. A recursive formula for $\{w_t^i)\}_{i=1,\dots,N}$ is derived. First, we demonstrate in appendix B that a recursive formula $p(X_{0:t}|Z_{1:t}^{1:S})$ is given by:

$$p(X_{0:t}|Z_{1:t}^{1:S}) \propto \prod_{s=1}^{S}\left\{\frac{p(X_t|Z_{1:t-1}^{1:S},Z_t^s)}{p(X_t|Z_{1:t-1}^{1:S})}\right\} \times p(X_t|X_{t-1})p(X_{0:t-1}|Z_{1:t-1}^{1:S}) \ . \quad (14)$$

This is the classical nonlinear fusion rule when there is a feedback step as given by Chong et al. in [3]. As in the previous section, we assume that the importance function is such that eq.(9) is satisfied. Then, incorporating eq.(14) and eq.(9) in eq.(6), we obtain:

$$w_t \propto \prod_{s=1}^{S}\left\{\frac{p(X_t|Z_{1:t-1}^{1:S},Z_t^s)}{p(X_t|Z_{1:t-1}^{1:S})}\right\} \times \frac{p(X_t|X_{t-1})}{\pi(X_t|X_{0:t-1},Z_{1:t}^{1:S})}w_{t-1} \ . \quad (15)$$

If we use $p(X_t|X_{t-1})$ as importance function (i.e. eq.(11)), then

$$w_t \propto \prod_{s=1}^{S}\left\{\frac{p(X_t|Z_{1:t-1}^{1:S},Z_t^s)}{\int p(X_t|X_{t-1})p(X_{t-1}|Z_{1:t-1}^{1:S})dX_{t-1}}\right\}w_{t-1} \ . \quad (16)$$

Finally, using the plug-in method,eq.(16) becomes

$$\boxed{w_t \propto \prod_{s=1}^{S}\left\{\frac{\hat{p}(X_t|Z_{1:t-1}^{1:S},Z_t^s)}{\int p(X_t|X_{t-1})\hat{p}(X_{t-1}|Z_{1:t-1}^{1:S})dX_{t-1}}\right\}w_{t-1}} \ . \quad (17)$$

Consequently, the particle filtering algorithm is composed of:

- a recursive formula for w_t given by eq.(17) which depends on $\{\hat{p}(X_t|Z_{1:t-1}^{1:S},Z_t^s)\}_{s=1,\dots,S}$ (i.e. information broadcasted by sensors at time t), $\hat{p}(X_{t-1}|Z_{1:t-1}^{1:S})$ (i.e. the global conditional estimate processed by the fusion node at time $t-1$) and $p(X_t|X_{t-1})$.
- trajectories $\{X_{0:t}^i\}_{i=1,\dots,N}$ sampled using $p(X_t|X_{t-1})$.

The distributed tracking algorithm is summed up in fig.3.

IV. Nonlinear fusion rules for local partial estimates

The aim of this section consists in deriving fusion rules in the context of nonlinear distributed tracking when sensor s broadcasts a local partial target state estimate. These fusion rules must depend on the prior information on the target and the information broadcasted by local sensors given in tab.II. First, let us remind that the aim is to estimate eq.(3). Using eq.(4), eq.(3) becomes:

$$\mathbb{E}\{f(X_{0:t})|Z_{1:t}^{1:S}\} = \int f(h(Y_{0:t}^{1:S}))p(Y_{0:t}^{1:S}|Z_{1:t}^{1:S})dY_{0:t}^{1:S} \quad (18)$$

Eq.(18) is a complex integral such that, as in the previous section, a particle filter approach is used to solve the problem. An importance function noted $\pi(Y_{0:t}^{1:S}|Z_{1:t}^{0:S})$ is introduced such that eq.(18) can be rewritten:

$$\mathbb{E}\{f(X_{0:t})|Z_{1:t}^{1:S}\} = \int f(h(Y_{0:t}^{1:S}))w_t\pi(Y_{0:t}^{1:S}|Z_{1:t}^{1:S})dY_{0:t}^{1:S} \quad (19)$$

where

$$w_t = \frac{p(Y_{0:t}^{1:S}|Z_{1:t}^{1:S})}{\pi(Y_{0:t}^{1:S}|Z_{1:t}^{1:S})} \ . \quad (20)$$

As in the previous section, the problem consists in sampling N trajectories $\{Y_{0:t}^{1:S,i}\}_{i=1,\dots,N}$ using the importance function and computing the weights $\{w_t^i\}_{i=1,\dots,N}$ such that:

$$\mathbb{E}\{f(X_{0:t})|Z_{1:t}^{1:S}\} \approx \sum_{i=1}^{N} f(h(Y_{0:t}^{1:S,i}))w_t^i \ . \quad (21)$$

We derive in the two following sections the particle filtering algorithm used as a fusion rule whether or not sensors architecture includes a feedback step.

A. Without feedback

We derive in this section the particle filtering algorithm used as a fusion rule when sensors architecture does not include feedback. A recursive formula for the weights $\{w_t^i\}_{i=1,\dots,N}$ is derived. We demonstrate in appendix C that:

$$p(Y_{0:t}^{1:S}|Z_{1:t}^{1:S}) \propto \prod_{s=1}^{S}\left\{\frac{p(Y_t^s|Z_{1:t}^s)}{p(Y_t^s|Z_{1:t-1}^s)}\right\} \times p(Y_t^{1:S}|Y_{t-1}^{1:S})p(Y_{t-1}^{1:S}|Z_{1:t-1}^{1:S}) \ . \quad (22)$$

We classically assume that the importance function is such that:

$$\pi(Y_{0:t}^{1:S}|Z_{1:t}^{1:S}) = \pi(Y_t^{1:S}|Y_{0:t-1}^{1:S},Z_{1:t}^{1:S})\pi(Y_{0:t-1}^{1:S}|Z_{1:t-1}^{1:S}) \ . \quad (23)$$

Then incorporating eq.(22) and eq.(23) in eq.(20), we obtain:

$$w_t \propto \prod_{s=1}^{S}\left\{\frac{p(Y_t^s|Z_{1:t}^s)}{p(Y_t^s|Z_{1:t-1}^s)}\right\} \times \frac{p(Y_t^{1:S}|Y_{t-1}^{1:S})}{\pi(Y_t^{1:S}|Y_{0:t-1}^{1:S},Z_{1:t}^{1:S})}w_{t-1} \ . \quad (24)$$

We use $p(Y_t^{1:S}|Y_{t-1}^{1:S})$ to sample trajectories:

$$\pi(Y_t^{1:S}|Y_{0:t-1}^{1:S},Z_{1:t}^{1:S}) = p(Y_t^{1:S}|Y_{t-1}^{1:S}) \quad (25)$$

such that eq.(24) becomes:

$$w_t \propto \prod_{s=1}^{S}\left\{\frac{p(Y_t^s|Z_{1:t}^s)}{\int p(Y_t^s|Y_{t-1}^s)p(Y_{t-1}^s|Z_{1:t-1}^s)dY_{t-1}^s}\right\}w_{t-1} \ . \quad (26)$$

Consequently, using the plug-in method, eq.(26) becomes:

$$\boxed{w_t \propto \prod_{s=1}^{S}\left\{\frac{\hat{p}(Y_t^s|Z_{1:t}^s)}{\int p(Y_t^s|Y_{t-1}^s)\hat{p}(Y_{t-1}^s|Z_{1:t-1}^s)dY_{t-1}^s}\right\}w_{t-1}} \ . \quad (27)$$

Now, let us precise the sampling of trajectories $\{Y_{0:t}^{1:S,i}\}_{\{i=1,\dots,N\}}$. We sample $\{Y_t^{1:S,i}\}_{\{i=1,\dots,N\}}$ from $\{Y_{t-1}^{1:S,i}\}_{\{i=1,\dots,N\}}$ using the following procedure given in fig.2. We can notice that we only use the prior information on target diffusion to generate these trajectories. Finally, the particle filtering algorithm is composed of:

- a recursive formula for w_t given by eq.(27) which depends on $\{\hat{p}(Y_t^s|Z_{1:t-1}^s)\}_{s=1,\dots,S}$,

$\{\hat{p}(Y_t^s|Z_{1:t-1}^s)\}_{s=1,\dots,S}$ (i.e. information broadcasted by sensors at time t and $t-1$) and $p(Y_t^s|Y_{t-1}^s)$ the prior information on partial target state diffusion.
- trajectories $\{Y_{0:t}^{1:S,i}\}_{i=1,\dots,N}$ are sampled using algorithm presented in fig.2.

The distributed tracking algorithm is summed up in fig.4.

B. With feedback

We derive in this section the particle filtering algorithm used as a fusion rule when sensors architecture includes a feedback step. A recursive formula for $\{w_t^i)\}_{i=1,\dots,N}$ is derived. We demonstrate in appendix D that a recursive formula $p(Y_{0:t}^{1:S}|Z_{1:t}^{1:S})$ is given by:

$$\begin{aligned} p(Y_{0:t}^{1:S}|Z_{1:t}^{1:S}) \quad &\propto \quad \prod_{s=1}^{S} \left\{ \frac{p(Y_t^s|Z_{1:t-1}^{1:S}, Z_t^s)}{p(Y_t^s|Z_{1:t-1}^{1:S})} \right\} \\ &\times \quad p(Y_t^{1:S}|Y_{t-1}^{1:S}) p(Y_{0:t-1}^{1:S}|Z_{1:t-1}^{1:S}) \ . \end{aligned} \tag{28}$$

We assume as in the previous section that the importance function is such that eq.(23) is satisfied. Then incorporating eq.(23) and eq.(28) in eq.(20), yields:

$$\begin{aligned} w_t \quad &\propto \quad \prod_{s=1}^{S} \left\{ \frac{p(Y_t^s|Z_{1:t-1}^{1:S}, Z_t^s)}{p(Y_t^s|Z_{1:t-1}^{1:S})} \right\} \\ &\times \quad \frac{p(Y_t^{1:S}|Y_{t-1}^{1:S})}{\pi(Y_t^{1:S}|Y_{0:t-1}^{1:S}, Z_{1:t}^{1:S})} w_{t-1} \ . \end{aligned} \tag{29}$$

Using eq.(25) as importance function, then eq.(29) becomes

$$w_t \propto \prod_{s=1}^{S} \left\{ \frac{p(Y_t^s|Z_{1:t-1}^{1:S}, Z_t^s)}{\int p(Y_t^s|Y_{t-1}^s) p(Y_{t-1}^s|Z_{1:t-1}^{1:S}) dY_{t-1}^s} \right\} w_{t-1} \ . \tag{30}$$

Consequently, using the plug-in method, eq.(30) becomes:

$$\boxed{w_t \propto \prod_{s=1}^{S} \left\{ \frac{\hat{p}(Y_t^s|Z_{1:t-1}^{1:S}, Z_t^s)}{\int p(Y_t^s|Y_{t-1}^s) \hat{p}(Y_{t-1}^s|Z_{1:t-1}^{1:S}) dY_{t-1}^s} \right\} w_{t-1}} \ . \tag{31}$$

Finally, the particle filtering algorithm is composed of:

- a recursive formula for w_t given by eq.(31) which depends on $\{\hat{p}(Y_t^s|Z_{1:t-1}^s, Z_t^s)\}_{s=1,\dots,S}$ (i.e. information broadcasted by sensors at time t), $\hat{p}(Y_t^s|Z_{1:t-1}^{1:S})$ (i.e. the global conditional estimate processed by the fusion node at time $t-1$) and $p(Y_t^s|Y_{t-1}^s)$ the prior information on partial target state diffusion.
- trajectories $\{Y_{0:t}^{1:S,i}\}_{i=1,\dots,N}$ are sampled using algorithm presented in fig.2.

The distributed tracking algorithm is summed up in fig.5.

V. Application to the bearings-only tracking problem

We present in this section the distributed bearings-only tracking problem and precise the nonlinear fusion equations derived in section III and IV.

A. The bearings-only tracking problem

We consider the problem of tracking a target using bearings measurements in the $x-y$ plane using S sensors . We note X_t the target state and $\mathcal{X}_t^s$ the state of sensor s such that:

$$\left\{ \begin{array}{l} X_t = \left[\begin{array}{cccc} r_x(t) & r_y(t) & v_x(t) & v_y(t) \end{array} \right]^* , \\ \mathcal{X}_t^s = \left[\begin{array}{cccc} \mathcal{R}_x^s(t) & \mathcal{R}_y^s(t) & \mathcal{V}_x^s(t) & \mathcal{V}_y^s(t) \end{array} \right]^* \ . \end{array} \right. \tag{32}$$

At each step of time, each sensor receives a bearing measurement. We note Z_t^s, the bearing measurement received at time t by sensors s. The target state is related to these measurements through the following equation:

$$Z_t^s = \arctan\left(\frac{r_x(t) - \mathcal{R}_x^s(t)}{r_y(t) - \mathcal{R}_y^s(t)}\right) + V_t^s \ , \quad \forall s \in \{1, \dots, S\} \tag{33}$$

where $V_t^s \sim \mathcal{N}(0, \sigma_v^2)$ and σ_v^2 is known. Moreover, it is assumed that the target follows a nearly constant-velocity model. The discretized state equation[1] is given by:

$$X_{t+1} = AX_t + \sigma W_t \ ,$$

where:

$$\left| \begin{array}{l} W_t \sim \mathcal{N}(0, Q) \ , \\ A = \left[\begin{array}{cc} 1 & \delta_t \\ 0 & 1 \end{array} \right] \otimes Id_{2\times 2} \ , \\ Q = \left[\begin{array}{cc} \frac{\delta_t^3}{3} & \frac{\delta_t^2}{2} \\ \frac{\delta_t^2}{2} & \delta_t \end{array} \right] \otimes Id_{2\times 2} \ . \end{array} \right. \tag{34}$$

δ_t is the elementary time period. The state covariance σ is known.

Considering a distributed tracking algorithm, each sensor sends a local conditional estimate of the full target state. The full target state estimate can be processed using a particle filtering algorithm using eq.(33) as measurement equation and eq.(34) as prior information on target diffusion. In this case, nonlinear fusion rules are given in section III. These formulas are precised in the following section. However, it is frequently argued that this algorithm is sometimes divergent and rather difficult to initialize [11] as the range is not always observable [10]. Another approach consists in processing only a partial state estimate at the sensor level given by:

$$Y_t^s = \left[\beta_t^s \quad \dot{\beta}_t^s \quad \ddot{\beta}_t^s\right]^* \ ,$$

where:

$$\beta_t^s = \arctan\left(\frac{r_x(t) - \mathcal{R}_x^s(t)}{r_x(t) - \mathcal{R}_x^s(t)}\right) \quad \forall s \in \{1, 2\} \ , \tag{35}$$

$\dot{\beta}_t^s$ and $\ddot{\beta}_t^s$ are the time derivatives of β_t^s. To perform tracking at the sensor level, we use the following measurement and diffusion equations:

$$\left\{ \begin{array}{l} Z_t^s = \beta_t^s + V_t^s , \\ Y_{t+1}^s = BY_t^s + \tilde{W}_t \ , \end{array} \right. \quad \forall s \in \{1, 2\}$$

[1] For a general review of dynamic models for target tracking see [15].

where:

$$\left| \begin{array}{l} V_t \sim \mathcal{N}(0, \sigma_v^2) \ , \\ \tilde{W}_t \sim \mathcal{N}(0, \Sigma) \ , \\ B = \begin{pmatrix} 1 & \delta_t & \delta_t^2 \\ 0 & 1 & \delta_t \\ 0 & 0 & 1 \end{pmatrix} \ . \end{array} \right. \tag{36}$$

σ_v is the measurement noise. Σ is a parameter which must be fixed. We can see that the stochastic system at the sensor level is a Gaussian linear one. Consequently, the Kalman filter is the optimal tracking algorithm. Local conditional mean and covariance estimates are sufficient statistics. In this case, nonlinear fusion rules given in section IV will be precised in subsection C.

B. Full state estimates are broadcasted by sensors

We have seen in the previous section that each sensor processes its local estimate using a particle filtering algorithm such that the PDF estimate is a sum of weighted Dirac delta functions noted:

$$\sum_{j=1}^{N} w_t^{s,j} \mathbb{1}_{X_t^{s,j}}(X_t) \tag{37}$$

where $X_t^{s,j}$ is the particle j of sensor s at time t and $w_t^{s,j}$ is the associated weight. However, in the case of bandwidth limitations, sensor s broadcasts only conditional mean and covariance estimates noted:

$$\hat{X}_t^s \text{ and } \hat{\sigma}_t^s \ . \tag{38}$$

The aim of this section is to precise nonlinear fusion rules presented in the section III in case whether the only estimated PDF is broadcasted or the only mean and covariance estimates are transmitted.

1) Low bandwidth, without feedback: If sensor s broadcasts conditional mean and covariance estimates i.e. $\hat{X}_t^s$ and $\hat{\sigma}_t^s$, then:

$$\left\{ \begin{array}{l} \hat{p}(X_t|Z_{1:t}^s) \propto e^{-\frac{1}{2}\|X_t - \hat{X}_t^s\|_{(\hat{\sigma}_t^s)^2}} \ , \\ \hat{p}(X_t|Z_{1:t-1}^s) \propto e^{-\frac{1}{2}\|X_t - A\hat{X}_{t-1}^s\|_{A(\hat{\sigma}_t^s)^2 A^* + Q}} \ . \end{array} \right. \tag{39}$$

Now incorporating eq.(39) in eq.(13), we obtain:

$$w_t \propto \prod_{s=1}^{S} \left\{ \frac{e^{-\frac{1}{2}\|X_t - \hat{X}_t^s\|_{(\hat{\sigma}_t^s)^2}}}{e^{-\frac{1}{2}\|X_t - A\hat{X}_{t-1}^s\|_{A(\hat{\sigma}_t^s)^2 A^* + Q}}} \right\} w_{t-1} \tag{40}$$

2) Low bandwidth, with feedback: If sensor s broadcasts mean and covariance estimates then

$$\hat{p}(X_t|Z_{1:t-1}^{1:S}, Z_t^s) \propto e^{-\frac{1}{2}\|X_t - \hat{X}_t^s\|_{(\hat{\sigma}_t^s)^2}} \ . \tag{41}$$

Moreover, the global estimated PDF at time $t-1$ is given by:

$$\hat{p}(X_{t-1}|Z_{1:t-1}^{1:S}) = \sum_{i=1}^{N} w_{t-1}^i \mathbb{1}_{X_{t-1}^i}(X_{t-1}) \ . \tag{42}$$

Now incorporating eq.(41) and eq.(42) in eq.(17), we obtain:

$$w_t \propto \prod_{s=1}^{S} \left\{ \frac{e^{-\frac{1}{2}\|X_t - \hat{X}_t^s\|_{(\hat{\sigma}_t^s)^2}}}{\sum_{i=1}^{N} p(X_t|X_{t-1}^i) w_{t-1}^i} \right\} w_{t-1} \ . \tag{43}$$

3) High bandwidth, without feedback: In this case, sensor s broadcasts the conditional PDF estimate:

$$\hat{p}(X_t|Z_{1:t}^s) = \sum_{j=1}^{N} w_t^{s,j} \mathbb{1}_{X_t^{s,j}}(X_t) \ . \tag{44}$$

Moreover, the local estimated PDF processed by sensor s at time $t-1$ is given by:

$$\hat{p}(X_{t-1}|Z_{1:t-1}^s) = \sum_{j=1}^{N} w_{t-1}^{s,j} \mathbb{1}_{X_{t-1}^{s,j}}(X_{t-1}) \ . \tag{45}$$

Now incorporating eq.(44) and eq.(45) in eq.(13), we obtain:

$$w_t \propto \prod_{s=1}^{S} \left\{ \frac{\sum_{j=1}^{N} w_t^{s,j} \mathbb{1}_{X_t^{s,j}}(X_t)}{\sum_{j=1}^{N} p(X_t|X_{t-1}^{s,j}) w_{t-1}^{s,j}} \right\} w_{t-1} \ . \tag{46}$$

4) High bandwidth, with feedback: In this case, sensor s broadcasts the local conditional PDF estimate:

$$\hat{p}(X_t|Z_{1:t-1}^{1:S}, Z_t^s) = \sum_{j=1}^{N} w_t^{s,j} \mathbb{1}_{X_t^{s,j}}(X_t) \ . \tag{47}$$

Moreover, the global estimated PDF at time $t-1$ is given by:

$$\hat{p}(X_{t-1}|Z_{1:t-1}^{1:S}) = \sum_{i=1}^{N} w_{t-1}^i \mathbb{1}_{X_{t-1}^i}(X_{t-1}) \ . \tag{48}$$

Now incorporating eq.(47) and eq.(48)in eq.(17), we obtain:

$$w_t \propto \prod_{s=1}^{S} \left\{ \frac{\sum_{j=1}^{N} w_t^{s,j} \mathbb{1}_{X_t^{s,j}}(X_t)}{\sum_{i=1}^{N} p(X_t|X_{t-1}^i) w_{t-1}^i} \right\} w_{t-1} \tag{49}$$

C. Partial target state estimates are broadcasted by sensors

In this case, we do not distinguish high and low bandwidth because the local conditional mean and covariance estimates noted:

$$\hat{Y}_t^s \text{ and } \hat{\sigma}_t^s \tag{50}$$

are sufficient statistics. However, we precise the recursive formulas whether or sensors architecture includes a feedback step.

1) Without feedback: If sensor s broadcasts conditional mean and covariance estimates i.e. $\hat{Y}_t^s$ and $\hat{\sigma}_t^s$, then

$$\left\{ \begin{array}{l} p(Y_t^s|Z_{1:t}^s) \propto e^{-\frac{1}{2}\|Y_t^s - \hat{Y}_t^s\|_{(\hat{\sigma}_t^s)^2}} \ , \\ p(Y_t^s|Z_{1:t-1}^s) \propto e^{-\frac{1}{2}\|Y_t^s - B\hat{Y}_{t-1}^s\|_{B(\hat{\sigma}_t^s)^2 B^* + \Sigma}} \ . \end{array} \right. \tag{51}$$

Now incorporating eq.(51) in eq.(27), we obtain:

$$w_t \propto \prod_{s=1}^{S} \left\{ \frac{e^{-\frac{1}{2}\|Y_t - \hat{Y}_t^s\|_{(\hat{\sigma}_t^s)^2}}}{e^{-\frac{1}{2}\|Y_t - A\hat{Y}_{t-1}^s\|_{B(\hat{\sigma}_t^s)^2 B^* + \Sigma}}} \right\} w_{t-1} \ . \tag{52}$$

2) With feedback: If sensor s broadcasts mean and covariance estimates then

$$p(Y_t^s|Z_{1:t-1}^{1:S}, Z_t^s) \propto e^{-\frac{1}{2}\|Y_t^s - \hat{Y}_t^s\|_{(\hat{\sigma}_t^s)^2}} \ . \tag{53}$$

Moreover the global estimated PDF is given by:

$$\hat{p}(Y_{t-1}^{1:S}|Z_{1:t-1}^{1:S}) = \sum_{i=1}^{N} w_{t-1}^i \mathbb{1}_{Y_{t-1}^{1:S,i}}(Y_{t-1}^{1:S}) \ , \tag{54}$$

so, that:

$$\hat{p}(Y_{t-1}^s|Z_{1:t-1}^{1:S}) = \sum_{i=1}^{N} w_{t-1}^i \mathbb{1}_{Y_{t-1}^{s,i}}(Y_{t-1}^s) \ . \tag{55}$$

Now, incorporating eq.(53) and eq.(55) in eq.(31), we obtain:

$$w_t \propto \prod_{s=1}^{S} \left\{ \frac{e^{-\frac{1}{2}\|Y_t^s - \hat{Y}_t^s\|_{(\hat{\sigma}_t^s)^2}}}{\sum_{i=1}^N p(Y_t^s|Y_{t-1}^{s,i}) w_{t-1}^i} \right\} w_{t-1} \ . \tag{56}$$

VI. Simulations results

Simulation results which illustrate distributed target tracking algorithms for the bearings-only tracking problem, as derived in the previous sections are presented in ftp://ftp.irisa.fr/techreports/2005/PI-1701.pdf.

VII. Conclusion

Nonlinear distributed target tracking for a single target has been addressed in this paper. We have investigated the case where sensors process local full/partial target state estimates. We have shown that the nonlinear fusion rules with/without feedback given by Chong et al. in [3] can be computed using particle filtering algorithms. Moreover, sensors architectures including low/high bandwidth have been discussed for the distributed for the bearings-only tracking setting.

Appendix A : proof of eq.(8)

This result is obtained applying Bayes formula, the Markovian property of the process $\{X_t\}_{t\in\mathbb{N}^+}$ and the independence of measurements given the target state. First, using Bayes formula, we have:

$$p(X_{0:t}|Z_{1:t}^{1:S}) \quad \propto \quad p(Z_t^{1:S}|X_t) p(X_{0:t}|Z_{1:t-1}^{1:S}) \ . \tag{57}$$

Now using the Markovian property of the process $\{X_t\}_{t\in\mathbb{N}^+}$, eq.(57) can be rewritten:

$$\begin{aligned} p(X_{0:t}|Z_{1:t}^{1:S}) \quad &\propto \quad p(Z_t^{1:S}|X_t) \\ &\times \quad p(X_t|X_{t-1}) p(X_{0:t-1}|Z_{1:t-1}^{1:S}) \ . \end{aligned} \tag{58}$$

Then, using measurements' independence given the target state, eq.(58) becomes:

$$\begin{aligned} p(X_{0:t}|Z_{1:t}^{1:S}) \quad &\propto \quad \prod_{s=1}^{S} p(Z_t^s|X_t) \\ &\times \quad p(X_t|X_{t-1}) p(X_{0:t-1}|Z_{1:t-1}^{1:S}) \ . \end{aligned} \tag{59}$$

Now, let us remark, using one more time Bayes formula that:

$$p(Z_t^s|X_t) = \frac{p(X_t|Z_{1:t}^s)}{p(X_t|Z_{1:t-1}^s)} p(Z_t^s|Z_{1:t-1}^s) \ . \tag{60}$$

Finally, incorporating eq.(60) in eq.(59), we obtain eq.(8).

Appendix B : proof of eq.(14)

Eq.(14) is obtained using the same ideas as in appendix A. First let us remark using Bayes formula that:

$$p(Z_t^s|X_t) = \frac{p(X_t|Z_{1:t-1}^{1:S}, Z_t^s)}{p(X_t|Z_{1:t-1}^{1:S})} p(Z_t^s|Z_{1:t-1}^{1:S}) \ . \tag{61}$$

Incorporating eq.(61) in eq.(59), we obtain eq.(14).

Appendix C : proof of eq.(22)

The set of partial target states $Y_t^{1:S}$ and the full target state X_t contain the same information. Then we can assume that $\{Y_t^{1:S}\}_{t\in\mathbb{N}^+}$ is a Markovian process and the independence of measurements given the set of partial target states. We use these two assumptions and Bayes formula to proove eq.(22). First, using Bayes formula, we have:

$$p(Y_{0:t}^{1:S}|Z_{1:t}^{1:S}) \propto p(Z_t^{1:S}|Y_t^{1:S}) p(Y_{0:t}^{1:S}|Z_{1:t-1}^{1:S}) \ . \tag{62}$$

Now using the Markovian property of the process $\{Y_t^{1:S}\}_{t\in\mathbb{N}^+}$, eq.(62) can be rewritten:

$$\begin{aligned} p(Y_{0:t}^{1:S}|Z_{1:t}^{1:S}) \quad &\propto \quad p(Z_t^{1:S}|Y_t^{1:S}) \\ &\times \quad p(Y_t^{1:S}|Y_{t-1}^{1:S}) p(Y_{0:t-1}^{1:S}|Z_{1:t-1}^{1:S}) \end{aligned} \tag{63}$$

Then, using measurements' independence given the target state, eq.(63) becomes:

$$\begin{aligned} p(Y_{0:t}^{1:S}|Z_{1:t}^{1:S}) \quad &\propto \quad \prod_{s=1}^{S} p(Z_t^s|Y_t^s) \\ &\times \quad p(Y_t^{1:S}|Y_{t-1}^{1:S}) p(Y_{0:t-1}^{1:S}|Z_{1:t-1}^{1:S}) \end{aligned} \tag{64}$$

Now let us remark, using one more time Bayes formula that:

$$p(Z_t^s|Y_t^s) = \frac{p(Y_t^s|Z_{1:t}^s)}{p(Y_t^s|Z_{1:t-1}^s)} p(Z_t^s|Z_{1:t-1}^s) \tag{65}$$

Finally, incorporating eq.(65) in eq.(64), we obtain eq.(22).

Appendix D : proof of eq.(28)

Eq.(14) is obtained using the same ideas as in appendix C. First, let us remark using Bayes formula that:

$$p(Z_t^s|Y_t^s) = \frac{p(Y_t^s|Z_{1:t-1}^{1:S}, Z_t^s)}{p(Y_t^s|Z_{1:t-1}^{1:S})} p(Z_t^s|Z_{1:t-1}^{1:S}) \ . \tag{66}$$

Incorporating eq.(66) in eq.(64), we obtain eq.(28).

References

[1] R. Singer and A. J. Kanyuck. Computer Control of Multiple Site Track Correlation. *Automatica*, 7:455–464, july 1971.

[2] Y. Zhu and X. Rong Li. Best Linear Unbiased Estimation Fusion. In *2th International Conference on Information Fusion, Sunnyvale, California, USA*, July 1999.

[3] C-Y. Chong, S.Mori, and K-C. Chang. Distributed Multitarget Multisensor Tracking. *in "Multitarget Multisensor Tracking", Artech House*, pages 248–295, 1990.

[4] Y. Zhu amd Z.you, J.Zhao, K. Zhang, and X. Rong Li. The Optimality for the Distributed Kalman Filtering Fusion with Feedback. *Automatica*, 37:1489–1493, 2001.

[5] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A Tutorial on Particule Filters for Online Non-Linear/Non-Gaussian Bayesian Tracking. IEEE *Transactions on Signal Processing*, 50(2), February 2002.

For t = 1 to T

1) **At the sensor level**: Computation of $\hat{p}(X_t|Z_{1:t}^s)$ using a tracking algorithm with:
   - *measurement equation:* $p(Z_t^s|X_t)$.
   - *prior information:* $p(X_{t+1}|X_t)$.
2) **At the fusion level**: Computation of $\hat{p}(X_t|Z_{1:t}^{1:S})$ using a particle filter:
   - For $i = 1, \dots, N$, sample $X_t^i \sim p(X_t|X_{t-1}^i)$,
   - Evaluate the weights using eq.(13),
   - Resampling step,

Fig. 1. *Distributed tracking with local full target state estimates without feedback*

1) Compute X_{t-1}^i using relation:
$$X_{t-1}^i = h(Y_{t-1}^{1:S,i}) \; .$$
2) Sample X_t^i using the prior information on target's diffusion i.e. $p(X_t|X_{t-1}^i)$.
3) Compute $Y_t^{s,i}$ for all s using the relation:
$$Y_t^{s,i} = g_s(X_t^i) \; .$$

Fig. 2. *the sampling of trajectories* $\{Y_{0:T}^{1:S,i}\}_{\{i=1,\dots,N\}}$.

[6] T.M. Nguyen, V.P. Jilkov, and X. Rong Li. Comparaison of Sampling-Based Algorithms for Multisensor Distributed Target Tracking. In *6th International Conference on Information Fusion, Cairns,Queensland, Australia*, July 2003.

[7] D. A. Castanon and D. Teneketzis. Distributued Estimation Algorithms for Nonlinear Systems. IEEE *Transactions on Automatic Control*, 31(5):418–425, may 1985.

[8] K.L. Anderson and R.A. Iltis. A Distributed Bearings-Only Traking Algorithm Using Reduced Sufficent Statistics. IEEE *Transactions on Aerospace, Electronic and Systems*, 32(1):339–349, january 1996.

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

[10] S.C. Nardone and V.J. Aidala. Observability Criteria for Bearings-Only Target Motion Analysis. IEEE *Transactions on Aerospace, Electronic and Systems*, 17:161–166, March 1981.

[11] T.Bréhard and J.-P. Le Cadre. Initialization of Particle Filter and Posterior Cramér-Rao Bound for Bearings- Only Tracking in Modified Polar Coordinate System. Technical report, IRISA, 2004.

[12] A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[13] A. Doucet. On Sequential Monte Carlo Sampling Methods for Bayesian Filtering. Technical report, Cambridge University Engineering Department, 2000.

[14] A. Doucet, N. De Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Pratice*, chapter Auxiliary Variable Based Particle Filters, pages 275–293. M.K. Pitt and N. Shephard, Springer edition, 2001.

[15] X. Rong Li and V. Jilkov. A Survey of Maneuvering Target Tracking Part I: Dynamics Models. IEEE *Transactions on Aerospace, Electronic and Systems*, 39(4):1333–1364, October 2003.

For t = 1 to T

1) **At the sensor level**: Computation of $\hat{p}(X_t|Z_{1:t-1}^{1:S}, Z_t^s)$ using a tracking algorithm with:
   - measurement equation: $p(Z_t^s|X_t)$.
   - prior information: $p(X_{t+1}|X_t)$.
2) **At the fusion level**: Computation of $\hat{p}(X_t|Z_{1:t}^{1:S})$ using a particle filter:
   - For $i = 1, \dots, N$, sample $X_t^i \sim p(X_t|X_{t-1}^i)$,
   - Evaluate the weights using eq.(17),
   - Resampling step,
3) Feedback

Fig. 3. *Distributed tracking with local full target state estimates with feedback*

For t = 1 to T

1) **At the sensor level**: Computation of $\hat{p}(Y_t^s|Z_{1:t}^s)$ using a tracking algorithm with:
   - measurement equation: $p(Z_t^s|Y_t^s)$.
   - prior information: $p(Y_{t+1}^s|Y_t^s)$.
2) **At the fusion level**: Computation of $\hat{p}(Y_t^{1:S}|Z_{1:t}^{1:S})$ using a particle filter:
   - For $i = 1, \dots, N$, sample $Y_t^{1:S,i}$ using fig.2 ,
   - Evaluate the weights using eq.(27),
   - Resampling step,

Fig. 4. *Distributed tracking with local partial target state estimates without feedback*

For t = 1 to T

1) **At the sensor level**: Computation of $\hat{p}(Y_t^s|Z_{1:t-1}^{1:S}, Z_t^s)$ using a tracking algorithm with:
   - measurement equation: $p(Z_t^s|Y_t^s)$.
   - prior information: $p(Y_{t+1}^s|Y_t^s)$.
2) **At the fusion level**: Computation of $\hat{p}(Y_t^{1:S}|Z_{1:t}^{1:S})$ using a particle filter:
   - For $i = 1, \dots, N$, sample $Y_t^{1:S,i}$ using fig.2 ,
   - Evaluate the weights using eq.(31),
   - Resampling step,
3) Feedback

Fig. 5. *Distributed tracking with local partial target state estimates with feedback*