



HAL
open science

Multiple target tracking with wireless sensor network for ground battlefield surveillance

B. Pannetier, J. Dezert, G. Sella

► **To cite this version:**

B. Pannetier, J. Dezert, G. Sella. Multiple target tracking with wireless sensor network for ground battlefield surveillance. FUSION 2014, Jul 2014, SALAMANQUE, Spain. hal-01070361

HAL Id: hal-01070361

<https://onera.hal.science/hal-01070361>

Submitted on 1 Oct 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multiple target tracking with wireless sensor network for ground battlefield surveillance

Benjamin Pannetier, Jean Dezert
ONERA

The French Aerospace Lab
Information processing and modelling department
F-91761 Palaiseau, France

Email: benjamin.pannetier@onera.fr, jean.dezert@onera.fr

Genevieve Sella
SAGEM

Defence and optic division, CED CV
100 avenue de Paris
91344 Massy cedex, France

Email: genevieve.sella@sagem.com

Abstract—For ground surveillance applications, the wireless sensor networks play a strategic role in military operations. In this paper, we explore the problem of tracking multiple targets observed in the sensors fields based on onboard algorithms under survivability system constraints. The surveillance system consists of a number of sensor nodes scattered in a region in order to detect and track targets in a cluttered environment. The targets can move on and off the road under several possible motion models. We study a Multiple Target Tracking (MTT) algorithm that fits with operational needs and offers good track continuity performance. The performances of this multiple sensors ground target tracking algorithm are evaluated on a complex and realistic scenario.

Index Terms—Wireless Ground Sensor Network, Multiple Ground Target Tracking, IMM algorithm

I. INTRODUCTION

The goal of the work presented in this paper is to study and develop in the next years an operational wireless sensor networks (WSN) which consists of large number of smart heterogeneous sensors with onboard sensing, processing and wireless communication capabilities for the French Ministry of Defense (MOD). The future operational WSN must satisfy severe exigencies in term of survivability (few months), low communications (to be undetectable by communication interception system), and real-time tactical situation assessment for large surveillance areas. The use of WSN network must also be easy and remotely controllable and have a low cost. The system must be easy to deploy, implemented by a limited number of operators with a minimum training through a simple human machine interface (HMI) for its exploitation and for decision-making support. Finally, the system must be modular, flexible and dynamically configurable (depending on the environment, the threat and mission). The main system characteristics of such system are:

- efficiency: the system must provide highest performances,
- modularity an operational flexibility,
- reliability: failures must be detected, isolated and substituted,
- real-time use: information must be received and preprocessed in real-time for the operational need,
- survivability: besides camouflage and discretion of the means deployed, optimizing the energy and the network

resistance to aggression is a problem for the operational credibility,

- affordability.

Components (both sensors and communications devices) must have low energy consumptions, to be able to work in a remote mode, in an outdoor environment and to fulfill discretion constraints required to work in unattended operating modes. The system must be easy to deploy and be able to adapt to various natures of terrain and topographies.

Our demonstrator is intended to allow studies on automatic data processing with an objective to correlate detection and generate only one alert on each target, being tracked as times goes on. It will allow us to evaluate several schemes for the data collection and fusion process, and to demonstrate the necessity of taking into account high-level information (typically geographic information, as traffic lanes, intersections, areas without terrain obscuration,...) for deployment and exploitation of the system.

Several processing levels are considered in this work:

- local processing of raw data at the sensor level: it can provide a detection alert on the presence of a target, and eventually some attributes about the target (as target location and type),
- additional processing on raw data (as basic image processing on sensor nodes),
- data fusion on a sensor node from a set of information collected from other sensors (target kinematics (e.g. tracks), classifications, their number, etc).

In this paper, we study the problem of tracking multiple moving objects observed through a WSN with limited sensing abilities. Our purpose is to track several targets in maintaining high track continuity performance to provide a reliable situation assessment. For this goal, we use heterogeneous sensors to compensate the low amount of data available (due to the weak sensor area coverage) by a better information quality on the data (both in precision of location and in classification information). The proposed data sensor processing presented in this work allows to meet the operational constraints.

Several papers have been published on operational sensor processing applied to WSN. For example, Ekman and Pålson

described in [1] a modified particle filter (PF) [2] to track a single vehicle through the WSN. Similar approach can be found in [3]. Despite of the well known estimation performances due to the generation of the particles on the road network, we haven't selected a PF algorithm because we need to track several targets in the sensor network with severe processing constraints due to hardware solution used in our demonstrator to preserve the power of a fusion node. In fact, because PF approach uses more CPU than Kalman filter (KF), extended Kalman filter (EKF) or unscented Kalman filter (UKF), we cannot use it in our specific context if one wants to make the surveillance system operational during a long period of time. Parmar and Zaveri in [4] have done similar studies and achieve the same conclusions. They focus their study of the data association for MTT in WSN and the need to limit the power to maintain the WSN in activity during a long time. However, in future work, if the hardware performances are improved in satisfying the power constraint the use of PF will become feasible. In fact, Oh and al. described in [5] a complete PF algorithm (called MCMCDA algorithm) applied for tracking multiple targets in a WSN with communication constraints. To improve the MTT algorithm performance, we introduce in this work the geographic informations in the tracking process as proposed by Ulmke and Koch in [6]. Since we are interested by tracking both ground vehicles (that can move on and off the road), aerial vehicles that are not constrained on the road, and pedestrians as well, we have to consider on-road tracking as well as an off-road tracking algorithms. For doing this, we have adapted the MTT ground target tracking algorithm described in [7] for our WSN tracking demonstrator.

The paper is organized as follows : in section II the WSN is briefly presented. Section III describes the multiple motion model algorithm constrained to geographic informations. Section IV is the extension of the algorithm to multiple target tracking with classification fusion. Results from the study are given in section V. Finally, concluding remarks are presented in section VI.

II. UNATTENDED GROUND SENSOR NETWORK

A. Network description

The good quality of communication between the sensor nodes has a strong impact on the ability of WSN to fulfill its task of surveillance. It is also very important that the WSN can communicate with the Command and Control (C2) station. The solution proposed in this paper is based on on-the-shelf existing components. Its multi-cluster architecture is represented in figure 1.

This architecture is structured in two levels:

- a set of clusters: sensor and fusion nodes connected through a low energy, low Rate 802.15.4 wireless network, managed by a gateway;
- a backbone with higher rate gathering data from clusters which guarantees the expected connectivity and allows two-ways communications.

The main information transmitted on the network are the following: data from sensor to sensor-nodes and to C2, state of

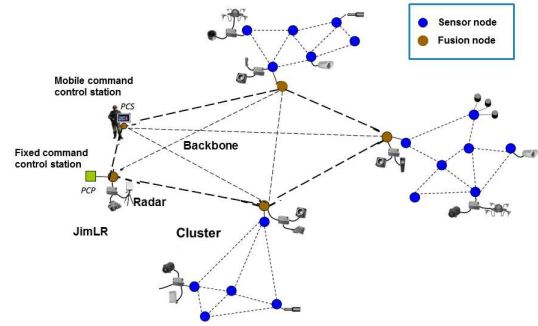


Figure 1: Sensor network architecture.

the components to sensor node and to C2, command to sensors from C2 or sensor node to components, exchange between sensor nodes to allow *horizontal* data fusion. Two categories of sensors: low consumption sensors that can be kept in operation to provide a continuous surveillance, and sensors having higher consumption that can be activated in case of presence of a target to acquire more detailed information on it.

The sensor node receives data from other sensors, processes them and transmits the local result to the fusion node. A set of complementary sensors is selected in order to collect multi-spectral information from the threats. These information will be used in order to

- detect the presence of a target, or an event,
- provide a spatial location of the event: sensors provide at current time t_k a measurement \mathbf{z}_k (bearing θ_k , elevation ϕ_k , distance ρ_k and radial velocity $\dot{\rho}_k$) in the sensor reference frame. Most sensors are able to give only partial location: bearing and distance only for radars, bearing and elevation for electro-optical sensors,
- classify the nature of event among the given set of classes C . The output of the classification process is a vector \mathbf{c}_k , where each component is the likelihood of each target class. Typically, we consider the following set of classes

$$C = \{light-vehicle, heavy-vehicle, tracked-vehicle, human, people, aerial targets\} \quad (1)$$

The classification class (project requirement) is not conventional because the class *heavy-vehicle* and *tracked-vehicle* are not exclusives. As well as the class *human* is include in the *people* (human group) class. The human class is a singleton of *people* class. That is why we have proposed at the sensor level for *light-vehicle* and *heavy-vehicle* classes to discriminate with sub-classes *tracked-light-vehicle*, *wheeled-light-vehicle* and *tracked-heavy-vehicle*, *wheeled-heavy-vehicle* respectively.

Different video algorithms have been studied at ONERA and we have integrated one of them in the sensor node to

detect, localize and classify automatically the targets with the previous considerations. The result of the processing (event, detection and classification information) is emitted to the fusion node. The same kind of process applies with acoustic sensors.

B. Sensor model

The generic sensor j observation model is given by:

$$\mathbf{z}_k^j = h^j(\mathbf{x}_k) + \mathbf{b}_k^j \quad (2)$$

where $h^j(\cdot)$ is the observation function, \mathbf{x}_k is the state of a target (detailed in the next section), and \mathbf{b}_k^j is a zero-mean white Gaussian noise vector with a known covariance matrix \mathbf{R}_k^j . The observation function and the associated noise depends on the type of sensor. We distinguish three observations functions: h^{radar} , h^{acou} , h^{optro} , h^{mag} associated respectively to the radar, acoustic, optic and magnetic sensors.

$$\begin{aligned} h^{radar}(\mathbf{x}_k) &= [\rho_k \quad \theta_k \quad \dot{\rho}_k]' h^{acou}(\mathbf{x}_k) = [\theta_k] \\ h^{opt}(\mathbf{x}_k) &= [\theta_k \quad \phi_k] h^{mag}(\mathbf{x}_k) = [x_k \quad y_k] \end{aligned} \quad (3)$$

For the magnetic sensor, we use its own location in the TCF in order to model a measurement because of its short range detection (see table I).

The different types of sensor that can be connected to a sensor node are listed in table I below. The *Volume* indicates the area coverage where the target can be found. This event is emitted as well as measurement to the fusion node in order to correlate this information with another volume, or a sensor detection to get a localized detection in the topographic coordinated frame (TCF).

Sensor type	number	sensor node output	detection characteristic
acoustic antenna	3	θ_k \mathbf{c}_k	spherical < 200 m
acoustic beacon	4	Volume \mathbf{c}_k	spherical < 200 m
magnetic	10	Volume \mathbf{c}_k	spherical < 2 m
radar	1	$\rho_k, \theta_k, \dot{\rho}_k$	Sectoral = 90° < 1000 m
PIR	8	θ_k	mono, multi beam < 200 m
micro-camera UIR	4	θ_k, ϕ_k \mathbf{c}_k	Sectoral = 30°, 40° < 100,200 m
FIR short UIR+visible	1	θ_k, ϕ_k \mathbf{c}_k	Sectoral = 10° < 100,200 m
JIM LR IR+visible	1	θ_k, ϕ_k \mathbf{c}_k	Sectoral = 10° < 100,200 m
Cham visible	4	θ_k, ϕ_k \mathbf{c}_k	Sectoral = 5°, to 50° < 100,200 m

Table I: Types of sensors used in the demonstrator.

C. Localization step

The localization module is used to localize all sensors and data in the TCF. For doing this, we need a calibration of each sensor. Several calibration techniques will be tested during the experimental trials, based on specific devices (GPS,

DGPS) allowing measurement of position and orientation of individual components, on cooperative localization using range or direction measurements between two sensors nodes, and on specific methods for calibration of electro-optics sensors.

For the sensors providing only volume information, or bearing detection, the localization module exploits all available information on sensors and elementary detection to provide a *composite report* $\mathbf{z}^{comp}(k)$ in the TCF that will feed the data fusion process. The sensors provide detections and information on location of the target in their own reference frame. To work in common TCF for situation assessment we always need a calibration step.

For notation convenience, the measurements sequence at the fusion node $Z^{k,l} = \{Z^{k-1,n}, \mathbf{z}_k^j\}$ represents a possible set of measurements generated by the target up to time k . $Z^{k,l}$ consists in a subsequence $Z^{k-1,n}$ of measurements up to time $k-1$ and a validated measurement \mathbf{z}_k^j available at time k from sensor j associated with the track $\mathcal{T}^{k,l}$. At the current time k , the track $\mathcal{T}^{k,l}$ is represented by a sequence of the state estimates.

III. TARGET TRACKING WITH GEOGRAPHIC INFORMATIONS

A. Geographic Information System

The geographic information system (GIS) used in this work contains the following information: the segmented road network, the hydrographic network, the vegetation area, the buildings area and DTED (Digital Terrain Elevation Data). Only the network and elevation information (the DTED + buildings height) are used in the first part of this study.

The road network is connected, and each road segment is indexed by the road section it belongs to. A road section is defined by a finite set of connected road segments delimited by a road end or a junction. For the topographic information, we use the database called: BD TOPO¹. This GIS has a metric precision on the road-segments location.

At the beginning of a surveillance battlefield operation, a TCF and its origin O are chosen in the manner that the axes X , Y and Z are respectively oriented in the East, North and Up local direction. The target tracking process is carried out in the TCF. In addition, starting from the elevation terrain and the sensor location at the current time, it is possible to compute the perceivability P_e at any referenced point for a sensor j . In the sequel, $P_e^j(x, y, k)$ will denote the probability for the sensor j to detect at time k a target at the location (x, y) .

B. Context constraint tracking

The target state at the current time t_k is defined in the local horizontal plane of the TCF by the vector:

$$\mathbf{x}_k = [x_k \quad \dot{x}_k \quad y_k \quad \dot{y}_k]^T \quad (4)$$

where (x_k, y_k) and (\dot{x}_k, \dot{y}_k) define respectively the target location and velocity in the local horizontal plane.

¹See www.professionnels.ign.fr/bdtopo for a description of this GIS.

The dynamics of the target evolving on the road are modeled by a first-order plant equation. The target state on the road segment s is defined by \mathbf{x}_k^s where the target position (x_k^s, y_k^s) belongs to the road segment s and the corresponding heading $(\dot{x}_k^s, \dot{y}_k^s)$ in its direction.

The event that the target is on road segment s is noted $e_k^s = \{\mathbf{x}_k \in s\}$. Given this event e_k^s and according to a motion model \mathcal{M}_i , the estimation of the target state can be improved by considering the road segment s . For a constant velocity motion model, it follows:

$$\mathbf{x}_k^s = \mathbf{F}^{s,i}(\Delta_k) \cdot \mathbf{x}_{k-1}^s + \mathbf{\Gamma}(\Delta_k) \cdot \mathbf{v}_k^{s,i} \quad (5)$$

where Δ_k is the sampling time, $\mathbf{F}^{s,i}$ is the state transition matrix associated to the road segment s and adapted to a motion model \mathcal{M}_i ; $\mathbf{v}_k^{s,i}$ is a white zero-mean Gaussian random vector with covariance matrix $\mathbf{Q}_k^{s,i}$ chosen in such a way that the standard deviation σ_d along the road segment is higher than the standard deviation σ_n in the orthogonal direction. It is defined by:

$$\mathbf{Q}_k^{s,i} = \mathbf{R}_{\theta_s} \cdot \begin{pmatrix} \sigma_d^2 & 0 \\ 0 & \sigma_n^2 \end{pmatrix} \cdot \mathbf{R}_{\theta_s}^T \quad (6)$$

where \mathbf{R}_{θ_s} is the rotation matrix associated with the direction θ_s defined in the plane (O, X, Y) of the road segment s . The matrix $\mathbf{\Gamma}(\Delta_k)$ is defined in [8].

To improve the modeling for targets moving on a road network, we have proposed in [9] to adapt the level of the dynamic model's noise based on the length of the road segment s . The idea is to increase the standard deviation σ_n defined in (6) to take into account the error on the road segment location. After the state estimation obtained by a Kalman filter, the estimated state is then projected according to the road constraint e_k^s . This step is detailed in [10].

C. IMM under road segment constraint

Here we recall briefly the principle of the interacting multiple model (IMM) taking into account the road network constraints. The IMM is a well-known efficient maneuvering target tracking algorithm [11] which combines estimated states based on multiple models to get a better global state estimate. The IMM is near optimal and has a reasonable complexity which makes it very appealing in tracking applications. In section III-B, a constrained motion model i to segment s , noted $\mathcal{M}_k^{s,i}$, was defined. There is a distinction between the definition of a motion model $\mathcal{M}_k^{s,i}$ (*i.e.* motion model type, noise,...) and the event $M_k^{s,i}$ that the target is moving on the road according the motion model i at time k . Here we extend the segment constraint to the different dynamic models (among a set of $r + 1$ motion models) that a target can follow. The model indexed by $r = 0$ is the stop model. The transition between the models is modelled as a Markovian process. In general when the target moves from one segment to the next, the set of dynamic models changes. In a conventional IMM estimator [11], the likelihood function of a model i is given, for a track $\mathcal{T}^{k,l}$, associated with the j -th measurement,

$j \in \{0, 1, \dots, m_k\}$ by:

$$\Lambda_k^i = p\{\mathbf{z}_k^j | M_k^{s,i}, Z^{k-1,n}\}, \quad i = 0, 1, \dots, r \quad (7)$$

where $Z^{k-1,n}$ is the subsequence of measurements associated with the track $\mathcal{T}^{k,l}$.

Using the IMM estimator with a stop-motion model, we get the likelihood function of the moving target mode for indexes $i \in \{0, 1, \dots, r\}$ and for $j \in \{0, 1, \dots, m_k\}$ by:

$$\Lambda_k^i = P_D \cdot p\{\mathbf{z}_k^j | M_k^{s,i}, Z^{k-1,n}\} \cdot (1 - \delta_{j,0}) + (1 - P_D) \cdot \delta_{j,0} \quad (8)$$

The likelihood of the stopped target mode (*i.e.* $r = 0$) is:

$$\Lambda_k^0 = p\{\mathbf{z}_k^j | M_k^{s,0}, Z^{k-1,n}\} = \delta_{j,0} \quad (9)$$

where $\delta_{j,0}$ is the Kronecker function defined by $\delta_{j,0} = 1$ if $j = 0$ and $\delta_{j,0} = 0$ otherwise.

The combined (global) likelihood function Λ_k of a track including a stop-motion model is then given by:

$$\Lambda_k = \sum_{i=0}^r \Lambda_k^i \cdot \mu_{k|k-1}^i \quad (10)$$

where $\mu_{k|k-1}^i$ is the predicted model probabilities. The steps of the IMM under road segment s constraint are the same as for the classical IMM and it has been described in [9].

Here, one has used the IMM algorithm constrained to only one road segment s . However, a road section is composed with several road segments. When the target is making a transition from one segment to another, the problem is to choose the segments with the corresponding motion models that can better fit the target dynamics. The choice of a segment implies the construction of the directional process noise. That is why the IMM motions model set varies with the road network configuration and a variable-structure IMM (VS IMM) offers a better solution for ground target tracking on road networks. Such algorithm has been denoted VS IMM (C standing for *Constrained*) and presented in details in [12].

D. Perceivability probability in the target tracking process

To maintain track continuity for improving the situation understanding and assessment for intelligence operation, we propose to study the non-detection causes of the sensors thanks to the knowledge one has (even partial) of the environment. The goal is to modify the likelihood of a track if the associated target is not detected, avoiding the stop-motion model activation and the stop of the track. A Bayesian formulation is proposed to introduce the target perceivability by the sensor in the likelihood (8). Based on our previous works [13], we introduce the event that the target associated with a track $\mathcal{T}^{k,l}$ is perceivable, or not, by the sensor j .

At time t_k , the target state probability is represented by the following exhaustive and exclusive events :

$$O_k^j = \{\text{target is perceivable by sensor } j\} \quad (11)$$

$$\bar{O}_k^j = \{\text{target is unperceivable by sensor } j\} \quad (12)$$

Here, O_k denotes the event that target can be detected by the sensor. By introducing the events (11) and (12) in the conventional IMM, we obtain a new formulation of the likelihood function. But the perceivability event does not take in account the non-detection due to the target stop. In the VS IMMC, we have for each motion model aside $r + 1$ motions models ($\forall i \in \{0, \dots, r\}$) the likelihoods function defined in (7) for a track $\mathcal{T}^{k,l}$. We recall that the track $\mathcal{T}^{k,l}$ represents the estimated states of the measurement sequence $Z^{k,l} = \{Z^{k-1,n}, \mathbf{z}_k^j\}$ with \mathbf{z}_k^j the detection of sensor j the current time t_k . Now, according to the total probability theorem, we introduce the event that the target is detected (*i.e.* $\{d = 1\}$) or not (*i.e.* $\{d = 0\}$) and the events O_k^j and \bar{O}_k^j . We obtain from (7) ($\forall i \in \{0, \dots, r\}$):

$$\begin{aligned} \Lambda_k^i &= p\{\mathbf{z}_k^j, d = 1, O_k^j | Z^{k-1,n}, M_k^{i,s}\} \\ &+ p\{\mathbf{z}_k^j, d = 1, \bar{O}_k^j | Z^{k-1,n}, M_k^{i,s}\} \\ &+ p\{\mathbf{z}_k^j, d = 0, O_k^j | Z^{k-1,n}, M_k^{i,s}\} \\ &+ p\{\mathbf{z}_k^j, d = 0, \bar{O}_k^j | Z^{k-1,n}, M_k^{i,s}\} \end{aligned} \quad (13)$$

However, an unperceivable target can't be detected. So the event $\{d = 1, \bar{O}_k^j\}$ is equal to the empty set \emptyset . According to Kirubarajan's approach [14], we distinguish the stop-motion model noted $M_k^{0,s}$ from the set of motion models. The event $\{M_k^{0,s}, d = 1\}$ is equal to \emptyset , because the stop-motion model must not be activated if there is at least one detection. By using Bayes' rule, we get the new expression of the likelihood function ($\forall i \in \{0, \dots, r\}$) as follows:

$$\begin{aligned} \Lambda_k^i &= (1 - \delta_{d,0}) \cdot P_D \\ &\cdot p\{\mathbf{z}_k^j | Z^{k-1,n}, M_k^{i,s}, O_k^j\} \cdot P\{O_k^j | Z^{k-1,n}, M_k^{i,s}\} \\ &+ (1 - P_D) \cdot \delta_{d,0} \cdot P\{O_k^j | Z^{k-1,n}, M_k^{i,s}\} \\ &+ \delta_{d,0} \cdot (1 - P\{O_k^j | Z^{k-1,n}, M_k^{i,s}\}) \end{aligned} \quad (14)$$

where $\delta_{d,0}$ is the Kronecker function equal to unity if there is no detection ($d = 0$). The probability to obtain at least one measurement is equal to the detection probability (*i.e.* $P\{d = 1 | Z^{k-1,n}, M_k^{i,s}, O_k^j\} = (1 - \delta_{d,0}) \cdot P_D$) in opposition to obtain no measurement (*i.e.* $P\{d = 0 | Z^{k-1,n}, M_k^{i,s}, O_k^j\} = (1 - \delta_{d,0}) \cdot (1 - P_D)$).

The computation of the perception probability $P\{O_k^j | Z^{k-1,n}\}$ depends on each sensor type, and it is computed at the location module. For instance, we use the line of site of camera type sensor to compute the non detection area due to terrain elevation with a ray tracing method. For the acoustic sensor, we use attenuation signal towards the ground nature and terrain elevation, etc. The perception probability is computed using the function P_e^j , more precisely by

$$P\{O_k^j | Z^{k-1,n}, M_k^{i,s}\} = P_e^j(x_{k|k-1}^{i,s}, y_{k|k-1}^{i,s}) \quad (15)$$

where $(x_{k|k-1}^{i,s}, y_{k|k-1}^{i,s})$ are the location components in the TCF of the predicted state $\hat{\mathbf{x}}_{k|k-1}^{i,s}$. In futures works, we will improve the computation of the perception probability using also the target velocity estimate.

IV. MULTIPLE TARGET TRACKING

A. Multiple target type tracker

We briefly describe here the main steps of the VS IMMC SB-MHT (Structured Branching - Multiple Hypotheses Tracking). More details can be found in chapter 16 of [8].

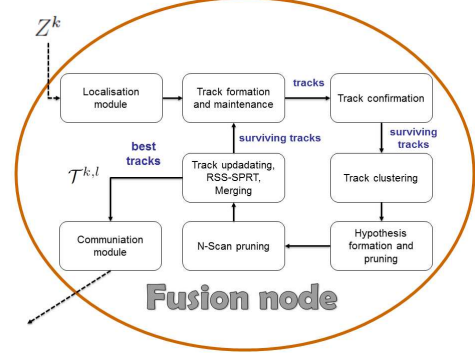


Figure 2: SB-MHT logic flowchart in a fusion node.

- 1) The first functional block of the SB-MHT is shown in figure 2. It consists of the track confirmation and the track maintenance. When the new set Z^k of measurements is received, a standard gating procedure [8] is applied in order to determine the valid measurement reports for track pairings. The existing tracks are updated with VS IMMC at first, and then extrapolated confirmed tracks are formed. When the track is not updated with reports, the stop-motion model is activated.
- 2) In order to palliate the association problem, we need a probabilistic expression for the evaluation of the track formation hypotheses that includes all aspects of the data association problem. It is convenient to use the log-likelihood ratio (LLR) as a score of a track $\mathcal{T}^{k,l}$ because it can be expressed at current time k in the following recursive form [8]:

$$L_{k,l} = L_{k-1,n} + \Delta L_{k,l} \quad (16)$$

with

$$\Delta L_{k,l} = \log \left(\frac{\Lambda_k}{\lambda_{fa}} \right) \quad (17)$$

and

$$L(0) = \log \left(\frac{\lambda_{fa}}{\lambda_{fa} + \lambda_{nt}} \right) \quad (18)$$

where λ_{fa} and λ_{nt} are respectively the false alarm rate and the new target rate per unit of surveillance volume. Λ_k is the global likelihood function described in (10). After the track score calculation of the track $\mathcal{T}^{k,l}$, Wald's Sequential Probability Ratio Test (SPRT) is used to set up the track status either as deleted, tentative or confirmed track. The tracks that fail the SPRT are deleted, and the surviving tracks are kept for the next stage.

- 3) The process of clustering is used to put altogether the tracks that share common measurements. The clustering

limits the number of hypotheses to generate, and therefore it can drastically reduce the complexity of tracking system. The result of the clustering is a list of tracks that are interacting. The next step is to form hypotheses of compatible tracks.

- 4) For each cluster, multiple compatible hypotheses are formed to represent the different compatible tracks scenarios. Each hypothesis is evaluated according to the track score function associated to the different tracks. Then, a technique is required to find the set of hypotheses that represents the most likely tracks collection. The unlikely hypotheses and associated tracks are deleted by a pruning method, and only the N_{Hypo} best hypotheses are kept in the system.
- 5) For each track, the *a posteriori* probability is computed, and a classical *N-Scan* pruning approach [8] is used to delete the most unlikely tracks. With this approach the most likely tracks are selected to reduce the number of tracks. However, the *N-Scan* technique combined with the constraint implies that other tracks hypotheses (*i.e.* constrained on other road segments) are arbitrary deleted. To avoid this problem, we modify the *N-Scan* pruning approach in order to select the N_k best tracks on each N_k road sections.
- 6) The SPRT is used to delete the unlikely hypotheses among the N_k hypotheses. The tracks are then updated and projected on the road network. In order to reduce the number of tracks to keep in the memory of the computer, a merging technique (selection of the most probable tracks which have common measurements) is also implemented.

B. Classification fusion

In a fusion node, the target type tracker presented in [15] is used to improve the performance of the data association in the SB-MHT. The principle consists to update the posterior class probability vector at each scan time t_k , with the classifier output. The classifier gives the probability vector $\beta_{k,l}$ of a track $\mathcal{T}^{k,l}$ given by :

$$\beta_{k,l} = \frac{\mathbf{c}_k^j \otimes \beta_{k-1,n}}{\mathbf{c}_k^{j'} \beta_{k-1,n}} \quad (19)$$

where \mathbf{c}_k^j is the likelihood vector of the j^{th} sensor classifier output, $\beta_{k-1,n}$ is the prior probability provided by the previous updated track $\mathcal{T}^{k-1,n}$ and \otimes is the Schur-Hadamard product. The initial classification vector is given by :

$$\beta_0 = \frac{\mathbf{c}_k^j}{\sum_{n=1}^{n=N} \mathbf{c}_k^j(n)} \quad (20)$$

In assuming the independence of the kinematic and classification observations, the augmented logarithm likelihood ratio $\Delta L_{k,l}^a$ is the sum of the logarithm kinematic-likelihood $\Delta L_{k,l}$ ratio given in (17), and the logarithm of classification ratio

$\Delta L_{k,l}^c$. The recursive form of the track score (16) is then given by

$$L_{k,l} = L_{k-1,n} + \Delta L_{k,l}^a \quad (21)$$

with

$$\Delta L_{k,l}^a = \Delta L_{k,l} + \Delta L_{k,l}^c \quad (22)$$

where $\Delta L_{k,l}$ is defined in (17).

The log-likelihood ratio of the classification belonging to the track $\mathcal{T}^{k,l}$ versus belonging to a false or new target is :

$$\Delta L_{k,l}^c = \log\left(\frac{\mathbf{c}_k^{j'} \beta_{k-1,n}}{\mathbf{c}_k^{j'} \beta_e}\right) \quad (23)$$

where e defines an extraneous target. If the track is not associated to a measurement at the current time t_k we have $\Delta L_{k,l}^c = 0$.

Finally, the updated target type $\hat{c}_{k,l}$ of the track $\mathcal{T}^{k,l}$ is chosen as the maximum probability of updated classification vector (19).

V. PERFORMANCE EVALUATION

To evaluate the performance of the proposed fusion process for WSN, we have simulated a realistic complex scenario. The goal of this evaluation is to prove the capability of our MTT algorithm to be implemented in a fusion node on-board prototype, and to provide measures of performance (MOP) of the tracking. For this, we have compared the proposed algorithm **with** and **without the road network information** respectively named **Algorithm 1** and **Algorithm 2**.

To compute the performance metrics, an important step is to decide at each time which track to compare with which target. In addition, this decision is made in the presence of closely spaced targets and false measurements. The assignment is required to be unique, *i.e.* at most one track can be associated with one target at any time, and at most one target can be associated with one track. To solve this assignment, Munkres algorithm has been used. The tracks not associated or correlated to a target despite the assignment are considered as false tracks.

The MOP that have been used in this study are the following:

- Root Square Error (RSE). The root square error is the most well-known MOP. It provides an information on the track precision in location and velocity.
- Track Length Ratio (TLR). The track length ratio is a ratio between the track length associated to a target with the length of the target trajectory. It informs on the track continuity performances

In our scenario, we have considered 20 targets moving on a chosen operational area. The targets are maneuvering on and off the road network. We distinguish several target types (as tank, jeep, soldiers, civilian pedestrians, etc). Our simulator constrained the maneuvers by taking into account the target type. We have had soldiers (targets number 11, 12, 13), and ground vehicles (targets number 14, 15, 19) that move on the

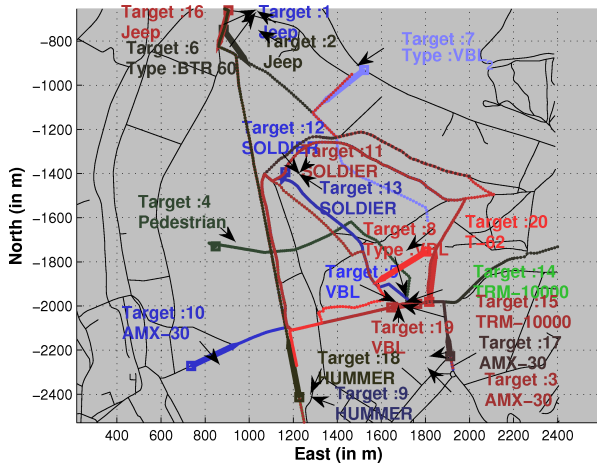


Figure 3: Ground target trajectories.

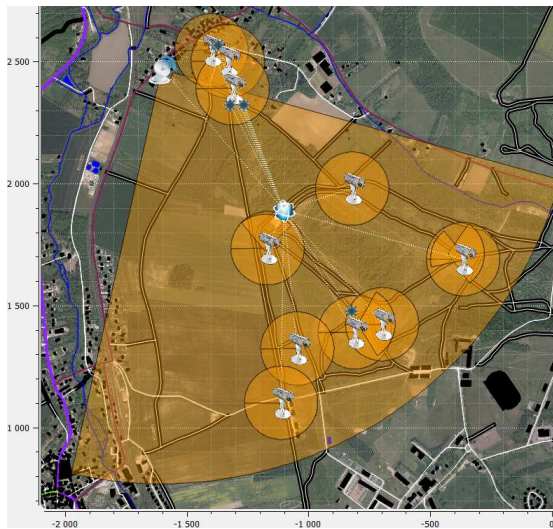


Figure 4: Sensor locations and coverages.

battlefield in a close formation. The figure 3 shows the targets trajectories on the area of interest. In this scenario, we also consider terrain masks, due to buildings, vegetation and terrain elevation.

Sensors² are placed at strategic locations (at some road intersections, or in order to get a maximum detection area) to ensure infrastructure protection mission. Radar, video and acoustic sensors generate false alarms.

To obtain our MOP we use only one simulation because our software is for operational vocation and is unable to integrated several simulations. The table II shows the Root Square Error (RSE) and the Track Length Ratio (TLR) of the **algorithm 1** (by tacking into account road network information) and **algorithm 2** (without road network information). Globally, we observe mitigated results on the track precision. In fact, we work with *in-situ* sensors with metric precision, that is why

²We are not allowed to give more details about sensors characteristics in this paper.

Target number	Algorithm 1		Algorithm 2	
	RSE (in m)	TLR	RSE (in m)	TLR
1	10.48	0.99	19.43	0.8
2	7.43	1	7.7	1
3	5.32	1	12	1
4	6.20	0.88	6.08	0.9
5	6.35	1	11.74	0.59
6	5.66	0.68	5.2	0.68
7	22.45	0.6	16.11	0.85
8	3.43	1	7.42	0.9
9	7.07	0.86	10.28	0.5
10	12.79	0.63	12.52	0.45
11	6.72	0.82	8.42	0.85
12	6.58	0.81	7.71	0.89
13	8.56	0.51	9.87	0.61
14	1.91	1	4.19	1
15	10.39	0.37	9.02	0.51
16	12.77	0.34	12.67	0.31
17	6.44	0.78	9.33	0.51
18	9.30	0.48	8.87	0.46
19	7.06	0.6	7.56	0.44
20	18.55	0.38	13.4	0.88

Table II: Synthetic MOP.

the improvement on the track precision, due to road network constraint, is weak. However, this constraint can provide better prediction on the track when the target evolves in non-detection areas (terrain mask or sensor absence) as shown on figures 5 and 6. The constraint contributes also to improve track association for the crossing maneuvers (figure 7). The cases where the **algorithm 2** has a better TLR than **algorithm 1** are due to the fact that the covariance is not constraint and less directional. The covariances of each motion model (necessary to the validation gating procedure) is more bigger than the covariances of each constrained motion model of the **algorithm 1**. The counterpart for **algorithm 2** is the cluster size, in the SB-MHT steps, which is bigger because more associations are done causing an increase of the computation time. But, in a strong target maneuver case out of the sensor coverage or in terrain mask, the **algorithm 1** is more robust to palliate the maneuver because the validation gate is larger. This is the case with the target 20 at the middle of the scenario, the target accelerates between 2 sensors area and the stop motion model is activated for the **algorithm 1** because no detection is associated at the opposite of the **algorithm 2** that succeeds in track-to-target association. A solution to compensate this weakness should be to compute a track segment association algorithm to correlate new tracks with with lost tracks . In addition, despite of the group class information given by video and acoustic sensors, algorithms don't arrive to track soldier's group. This is due to the heterogeneous measurement model. A group is only one detection for the previous sensors brings about a track initialisation. But with radar sensor a group can be several detections due to resolution cell. An ambiguity arises in track association if several heterogeneous sensor detect a group resulting track lost. The on-boarded constraints (not communicated in this paper) are satisfied.



Figure 5: Track leaving the video coverage.



Figure 6: Track continuity is maintained.

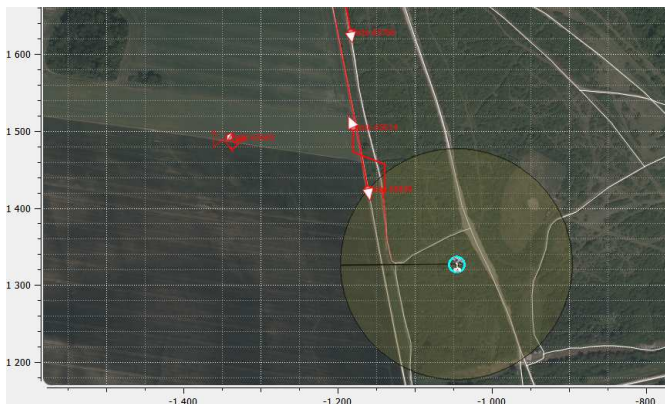


Figure 7: Track crossing illustration.

VI. CONCLUSION

To conclude, this paper presents well-known algorithms applied for wireless sensor network with severe on-boarded constraints and operational requirements. We have described a multiple target tracking algorithm in a fusion node and validated the concept on a simulated scenario. The main weakness of our approach is the lost of tracks when the targets evolve in close formation. The next steps of our project are : to test the fusion node in operational context with real sensors and associated processing, to develop an approach to initialise and track groups in a heterogeneous sensor network, to compare the performances to use distributed and hierarchical data fusion architecture in order to limit the bandwidth and to propose finally an approach to detect abnormal behaviour with a WSN activated during several weeks on a operational theatre.

ACKNOWLEDGMENT

The authors thank the French MOD for the financial and operational support of this research work.

REFERENCES

- [1] M. Ekman, H. Palsson, *Ground target tracking of vehicles in a wireless sensor network*, in Proc. of Fusion 2012 conf., Singapore, 2012.
- [2] M. Ekman, E. Sviestins, *Mutiple model algorithm based on particle filters for ground target tracking*, in Proc. of Fusion 2007 Conf., Québec, July 2007.
- [3] D. Salmond, M. Clark, R. Vinter and S. Godsill, *Ground target modelling, tracking and prediction with road networks*, in Proc. of Fusion 2007, Conf., Québec, July 2007.
- [4] P. Parmar, M. Zaveri, *Mutiple target tracking and data association in a wireless sensor network*, in Proc. of 4h int. conf. on computational intelligence and communications network, November 2012.
- [5] S. Oh, S. Sastry and L. Shenato, *A hierarchical multiple-target tracking algorithm for sensor networks*, in Proc. of the 2005 IEEE int. conf. on robotics and automation, Barcelona, Spain, April 2005.
- [6] M. Ulmke, W. Koch, *Road-map assisted ground moving target tracking*, IEEE Trans on AES, Vol. 42, pp. 1264–1274, October 2006.
- [7] B. Pannetier, J. Dezert, *Track segment association with classification information*, Workshop on Sensor Data Fusion: Trends, Solutions, Applications (SDF), 2012.
- [8] S.S. Blackman, R. Popoli, *Design and analysis of modern tracking systems*, Artech House, Design and analysis of modern tracking systems, 1999.
- [9] B. Pannetier, J. Dezert and E. Pollard, *Improvement of Multiple Ground Targets Tracking with GMTI Sensor and Fusion of Identification Attributes*, IEEE Aerospace Conf., Mar. 2008.
- [10] C. Yang, E. Blasch *Kalman Filtering with Nonlinear State Constraints*, IEEE Trans. on AES, Vol. 45, No. 1, pp. 70–84, 2009.
- [11] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, John Wiley and Sons, New York, Chapter 11, pp. 421–490, 2001.
- [12] B. Pannetier, V. Nimier and M. Rombaut, *Multiple ground target tracking with a GMTI sensor*, in Proc. of MFI 2006, Sept. 2006.
- [13] J. Dezert, N. Li, and X.R. Li, *A new formulation of IPDAF for tracking in clutter*, in Proc. Euro. Contr. Conf., Karlsruhe, Germany, Sept. 1999.
- [14] T. Kirubarajan and Y. Bar-Shalom, *Tracking evasive move-stop-move targets with an MTI radar using a VS-IMM estimator*, IEEE Trans. on AES, Vol. 39, No. 3, pp. 1098–1103, 2003.
- [15] Y. Bar-Shalom, T. Kirubarajan and C. Gokberk, *Tracking with classification-aided multiframe data association*, IEEE Trans. on AES, Vol. 41, No. 3, pp. 868–878, 2005.