A multiple-hypotheses map matching method suitable for weighted and box-shaped state estimation for localization

Fahed Abdallah, Ghalia Nassreddine and Thierry Denœux

***Abstract*—The goal of map-matching algorithms is to identify the road taken by a vehicle and to compute an estimate of the vehicle position on that road using a digital map. In this paper, a map-matching algorithm based on interval analysis and belief function theory is proposed. The method combines the outputs from existing bounded error estimation techniques with piecewise rectangular roads selected using evidential reasoning. A set of candidate roads is first defined at each time step using the topology of the map and a similarity criterion, and a mass function on the set of candidate roads is computed. An overall estimate of the vehicle position is then derived after the most probable candidate road has been selected. This method allows multiple road junctions hypotheses to be handled efficiently, and can cope with missing data. Also, the implementation of the method is quite simple as it is based on geometrical properties of boxes and rectangular road segments. Experiments with simulated and real data demonstrate the ability of this method to handle junction situations and to compute an accurate estimate of the vehicle position.**

***Index Terms*—Map-matching, Dempster-Shafer theory, Evidential reasoning, Multi-sensor fusion, Interval analysis, Bounded error estimation, Multiple Hypotheses technique.**

I. Introduction

Many research and industrial applications require accurate localization and/or tracking of moving vehicles [14][26][42]. Satellite-based navigation systems like the global positioning system (GPS) are playing an important role in vehicle localization as a result of their 24-hour, all weather, free-of-charge availability. However, GPS by itself is not always the ideal solution. It was originally designed with an inherent error of at least 10 meters for non-military applications. Also, it suffers from line of sight issues that make it less effective in urban canyons. One possible solution for correcting GPS error is the integration of GPS data with other sensor data such as dead reckoning (DR) using a data fusion algorithm [48].

A critical issue when designing data fusion algorithms is the representation of uncertainties that pervade both sensor measurements and the state space model. Two main categories of data fusion algorithms can be distinguished:

- Recursive Bayesian estimation, based on a probabilistic description of uncertainty;
- State bounding methods, based on a set-membership description of uncertainty.

Methods in the first category assume the measurement noise and state perturbations to be realizations of random variables with known statistical properties. This is the mainstream approach. The most common approaches are the Extended Kalman filter (EKF) and the particle filter (PF). The EKF linearizes the state and measurement equations and then applies the Kalman filter to obtain state estimates [8][21], assuming the process and observation noises to be normal. The state posterior probability distribution is approximated by a Gaussian distribution that is propagated analytically through the linearized system equations. However, the linearization is inherently local and may fail to produce reliable estimates, especially when the state model is highly non-linear. The statistical interpretation of covariance matrices is also unclear in this approach, as the statistical properties of the perturbations though non-linear equations is ill-known.

Sequential Monte Carlo methods for recursive Bayesian filtering, often referred to as particle filters (PF), usually provide more accurate information about the state posterior probability distribution than does the EKF, especially if it has a multimodal shape or if the noise distributions are non Gaussian [41]. The efficiency and accuracy of the PF depends mainly on the number of particles and on the propagation method used to re-allocate weights to these particles at each iteration. To cope with high measurement uncertainty, a large number of particles has to be used, especially when the dimension of the state vector is large; this may be an issue for real time implementation of the PF. Several authors have tried to overcome these shortcomings by combining approaches (see [25] and references therein) or by adapting the size of sample sets during the estimation process [19].

The classical data fusion algorithms mentioned above are strongly affected by several types of measurement errors like bias and drift, or even by partial or total conflict between the sources of information [4][6][10]. Moreover, these methods critically rely on accurate state space and observation noise models, which are rarely available.

Methods in the second group, referred to as *state bounding* or *bounded error* methods, assume all variables to belong to known compact sets and attempt to build simple sets, such as boxes, guaranteed to contain all state vectors consistent with given constraints [29]. These methods are deterministic approaches and handle only interval information acquired from multiple sources. The interval framework has been shown by several authors to be a good methodology for dealing with non-white and biased measurements [1][7][23][29][30].

Recently, a relatively simple and fast method based on constraint propagation and interval analysis has been introduced in [22][23] with an application to vehicle localization. The use of the box representation of the state is dictated by computational convenience: the main advantage of this representation is that intervals can easily be manipulated using interval analysis. The main interest of the bounded-error approach arises from the fact that it allows so-called *validated computations*, i.e., computations with guaranteed accuracy taking into account all possible sources of error, from data imprecision to rounding errors due to computer calculations. The major drawback of this approach is the difficulty in determining noise bounds. If the bounds are too tight, the data may become inconsistent with the system equations, in which case the method fails to provide a solution, but if the bounds are overestimated, the estimated state becomes very imprecise, and the method becomes overly pessimistic.

Integrated GPS/DR systems using data fusion usually fail to provide the current vehicle position on a given road segment [2][3][9][47]. The availability of an accurate digital road network makes it possible to find the vehicle position on a road segment. This technique is referred to as *map matching* (MM). The general purpose of an MM algorithm is to identify the correct road segment on which the vehicle is travelling and to compute the vehicle position on that segment. A number of different algorithms have been proposed for MM in different applications [11][20][28][49][50]. In most existing MM algorithms, the parameters used to select a precise road segment are based on the proximity between the vehicle position and the road, the degree of correlation between the vehicle trajectory and the road centerline, and the topology of the map. In [28] and [35], the authors proposed a MM algorithm in which the multiple hypotheses technique (MHT) is used in order to compute a more accurate estimate of the vehicle position on the road segment. MHT involves keeping track of several positions of the vehicle simultaneously, and, if possible, selecting the best candidate. In this approach, probability theory is used for identifying candidate roads and for selecting the best one [27], [35].

In this paper, we propose an MM algorithm called Belief Map Matching (BMM), that combines bounded error state estimation with a rectangular road map representation and that manages multiple hypotheses using *belief function theory*, also referred to as *Dempster-Shafer theory* [32], [39], [45], [51]. The rectangular road representation takes geometrical errors in the map and a priori information on road width into account. This representation makes it possible to combine map information with box state estimates in a convenient way. A belief function on a set of candidate roads is computed using geometrical and topological information provided by the map. This method allows us to handle multiple hypotheses at junctions or when roads are parallel, and to cope with missing data.

The paper is organized as follows. An overview of existing map-matching algorithms is first presented in Section II. Section III introduces the background on basic tools used in this work, i.e., interval analysis and belief function theory. Our method is then introduced in Section IV. Experimental results with simulated and real data are presented in Section V. Finally, Section VI summarizes and discusses the main contributions of the paper.

II. Review of Map-Matching Algorithms

The goal of map-matching (MM) algorithms is to identify the road taken by a vehicle and to compute an estimate of the vehicle position on that road using a digital map. In this section, we present an overview of existing MM methods and we discuss their limitations as well as the motivations of our work.

A. Overview of Existing MM methods

MM algorithms can be classified into three categories. Algorithms in the first category consider only the geometric relationships between the estimated position of the vehicle and a digital map [47][50]. Algorithms in the second category also consider the topology of the road network and historical data regarding the estimated position of the vehicle [24]. Finally, methods in the third category, referred to as *advanced methods*, use sophisticated tools such as the Kalman Filter (KF) [28], Bayesian networks [43], Dempster-Shafer theory (also known as Belief theory) [17] or fuzzy logics [46].

The most common geometric MM approach is based on a simple search concept [36]. In this approach, the vehicle position is matched to the nearest node or shape point in the digital map. This approach is known as *point-to-point matching* [9]. It is easy to implement, but is strongly affected by the way in which the network was digitized [36]. Another geometric MM approach is *point-to-curve matching* [47][50]. In this approach, the vehicle position is matched to the nearest road (curve) in the road network. As each road is composed of several line segments, the distance between the vehicle position and each line segment is calculated. The line with the smallest distance is judged to be the one on which the vehicle is travelling. This approach is usually more efficient than point-to-point matching but it generates unstable results in dense urban networks [36]. *Curve to curve matching* constitutes yet another geometric MM approach. In this approach, the candidate nodes are first selected using point-to-point matching. Then, for each candidate node, piecewise linear curves are constructed from the set of paths that originate from that node. Also, a piecewise linear curve can be constructed using the vehicle trajectory. The distance between the trajectory and the road curve is then calculated. The road at the smallest distance from the vehicle trajectory is selected. Unfortunately, this approach is quite sensitive to outliers and it sometimes yields unexpected and undesirable results [36].

In the second category of algorithms, in addition to geometric information, the result of the MM algorithm at time step $k-1$ is used for selecting candidate roads at time step k using the topology of the road network as a constraint [47]. These methods are usually based on a weighted topological algorithm for selecting the vehicle's current road. Using the topology analysis of a road network and the coordinates or the trajectory information of the vehicle, this algorithm can compute the weight of each candidate road. The actual road taken by the vehicle is judged to be the one with the largest weight. However, if the result given by the algorithm was wrong at time step $k-1$, then the result at time step k is also likely to be wrong, especially after a junction or where there are close parallel roads. This constitutes a major implementation problem and drawback of this method.

We may remark that most existing map matching algorithms apply a simple perpendicular projection of the position fixes onto the selected links, and ignore the numerous errors associated with the positioning sensors and the spatial road network data.

Advanced MM methods are based on more elaborate tools such as the KF, Demspter-Shafer theory, fuzzy logics and the MHT [17][31][35][46].

In [17], a road selection method based on multi-criteria fusion using belief function theory is proposed. In this method, a confidence region is used for selecting a set of candidate roads (CRs). Proximity and angular criteria are used in order to define a belief function on a set of CRs. The road with the highest degree of belief is then selected. This method, however, can yield inaccurate results in the case of parallel roads, as it does not consider the topology of the road network when selecting the set of CRs. Also, when GPS measurements are not available, the estimated position can be attributed to the wrong road, as this method does not use the multiple hypotheses technique.

In [31], an MM method based on the KF and geometrical properties is presented. In this method, the most plausible road is chosen using the point-to-curve method. The orthogonal projected location of the position fix onto the link is used as the initial vehicle location. Due to the point-to-curve projection, the cross-track error is reduced, as compared to the point-to-point method. However, the along-track error remains a key issue. A KF is then designed to re-estimate the vehicle position with the objective of minimizing the along-track error. As stated above, the point-to-curve method is not sufficient for selecting the correct link, especially in dense urban road networks. If the identification of the link is incorrect, then the inputs of the KF will also be inaccurate, which may result in further positioning errors.

In [46], the authors describe a map matching algorithm based on a fuzzy model. This algorithm is composed of two main steps. The first step, referred to as the *first fix mode*, is the initialization step. It is based on a fuzzy inference system (FIS), which is used to identify the correct road of the vehicle. The FIS selects a set of links that are within a given region of the GPS/DR position fix. A link is then identified based on the direction of the vehicle relative to the direction of the links and the heading change from the gyroscope. The location of the vehicle is then determined by an orthogonal projection of the position fixed onto that link. This step takes about 30 sec to identify the first road. In the second step, called the *tracking mode*, another FIS is used to find out whether the subsequent position fixes can be matched to the link identified in the first fix mode. The inputs are proximity, orientation and distance traveled by the vehicle along the link. If there are any outliers in the GPS/DR outputs, or if a turn is detected, then the algorithm goes back to the first fix mode. This method is not recommended in urban areas. Indeed, the algorithm

TABLE I

NEEDED INFORMATION FOR USED MM METHODS

Used information	[36]	[17]	BMM
Punctual state information	Yes	Yes	No
Bayesian state representation	No	Yes	No
Bounded state representation	No	No	Yes
Geometrical information	Yes	Yes	Yes
Roads connectivity	No	No	Yes
Similarity in orientation	Yes	Yes	No
Roads width	No	Yes	Yes

may have to use the first fix mode frequently, which induces a delay in the identification of the first correct link. Also this method does not take into account the error sources associated with the navigation sensors and the digital maps and hence the determination of vehicle location is not robust.

In [35], a map matching algorithm using MHT is proposed. The main contribution of this work is the use of MHT to solve the MM problem in some crucial situation such as junctions and parallel roads. MHT, which uses measurements from a validation region, is reformulated as a single target problem to develop the map matching method. Pseudo-measurements are generated for all links within the validation region defined as the error ellipse from the GPS/DR system. Pseudo-measurements (position and heading) are defined as the projected points of the GPS/DR positions on the links. The topological analysis of the road network (connectivity, orientation, and road design parameters) together with the pseudo-measurements are used to derive a set of hypotheses and their probabilities for each GPS/DR sensor output. Hypotheses with probabilities below a certain threshold are rejected. The map matching is only applied to the valid hypotheses and then a KF is executed to estimate the vehicle position on the CR corresponding to each hypothesis. The estimated position corresponding to the hypothesis with the highest probability is selected as the most plausible position of the vehicle.

B. Motivations of this Work

Map matching methods are based on combining an available state estimate with data extracted from a map database. These methods differ according to whether they are applied to probabilistic or bounded state estimates, to the subset of attributes used from the available map database, and to how these data are merged.

Classical MM methods rely on a Bayesian representation of the state uncertainty and are not suitable for handling bounded error estimation.

Box or bounded representation of the state has proved to be efficient in many applications, as shown in [1][30]. As a consequence, a new map matching method based on this type of state representation is becoming increasingly necessary. This in fact is one of the principle motivations of the BMM method presented in the paper. Furthermore, using a state box representation seems to be more intuitive when dealing with a rectangular road representation. The Multiple hypothesis assumption of the MM problem constitutes one of the advantages of the method. Depending on the estimation approach, criteria based on the proximity, historical navigation information and the topology of the road network have to be adapted.

One major problem of some existing MM methods is that they do not consider exhaustively the map error and the road width in the selection of the correct link and in the determination of the vehicle position. In the proposed MM method, a rectangular representation of the roads is used to overcome the problem of existing MM method by taking into account in a natural way map errors and road widths, which are quantities available in recent map databases. Furthermore, the rectangular representation of the map allows us to take into account several types of error on the available digital map, as it is more exhaustive than other representations used in some existing works [28][18]. In addition, state boxes and rectangular road representations fulfil naturally the integrity requirements studied in [37]. This is explained in more details in Section IV-E.

Belief function theory will be used for road selection and to handle multiple hypotheses. As will be shown, the use of belief functions makes it possible to detect missing map data (a common problem in MM) in a simple and efficient way. In contrast, this issue requires more complex procedures in existing classical map matching methods [11][28]. Furthermore, the management of multiple hypothesis cases (junction and parallel road situations) requires a complicated implementation under the Bayesian framework because of combinatorial problems [28]. As will be shown, the theory of belief functions makes it possible to design a simple solution to this problem, using the topology of the map and historical navigation information (link ID, vehicle location, vehicle heading and speed or elementary displacement).

Table I set out the information requirements of the different methods to be implemented in this paper. This table clearly shows that only our method is well adapted to bounded error approaches. Also, this method gathers simultaneously, and in a simple manner, different topological attributes of the map database. Note that even though the similarity in orientation criterion is not used in the BMM method, the flexibility of the belief function theory used to merge information makes it easy to take into account this criterion (or any other criterion) in the BMM algorithm.

Table II summarizes the ideas behind some MM algorithms that are discussed in the paper. This table also shows the advantages and the limitations of the different algorithms.

III. Basic Tools

In this section, the basic tools used in this paper are briefly introduced. Interval analysis is first presented in Subsection III-A, and the necessary background on belief functions is recalled in Subsection III-B.

A. Interval Analysis

We briefly present interval analysis and we describe the constraint propagation technique, also referred to as the consistence technique by some authors.

1) Basic Notions: Usually, interval analysis is used to model quantities that vary around a central value within certain bounds. A real interval, denoted $[x]$, is defined as a closed and connected subset of $\mathbb{R}$:

$$[x] = [\underline{x}, \overline{x}] = \{x \in \mathbb{R} / \underline{x} \leq x \leq \overline{x}\},$$

where $\underline{x}$ and $\overline{x}$ are the lower and upper bounds of $[x]$. A box $[\mathbf{x}]$ of $\mathbb{R}^n$ is defined as a Cartesian product of n intervals:

$$[\mathbf{x}] = [x_1] \times [x_2] \cdots \times [x_n].$$

The set of n-dimensional boxes will be denoted by $\mathbb{IR}^n$.

All set-theoretic operations, e.g., $\cap, \cup, \ldots$ as well as the four elementary arithmetic operations $\{+, -, *, /\}$ can be extended to the interval context. In general, the image of a box $[\mathbf{x}] \in \mathbb{IR}^n$ by a function $\mathbf{f}$ may have any shape. It may be non-convex shape or even multiply connected if $\mathbf{f}$ is discontinuous, as shown in Figure 1. An interval function $[\mathbf{f}]$ from $\mathbb{IR}^n$ to $\mathbb{IR}^m$ is said to be an *inclusion function* for $\mathbf{f}$ if

$$\mathbf{f}([\mathbf{x}]) \subseteq [\mathbf{f}]([\mathbf{x}]), \quad \forall [\mathbf{x}] \in \mathbb{IR}^n. \tag{1}$$

Inclusion functions may be very pessimistic, as shown by Figure 1. The inclusion function $[\mathbf{f}]$ is minimal if, for any $[\mathbf{x}]$, $[\mathbf{f}]([\mathbf{x}])$ is the smallest box of $\mathbb{R}^m$ containing $\mathbf{f}([\mathbf{x}])$. The minimal inclusion function for $\mathbf{f}$ is unique and may be denoted $[\mathbf{f}]^*$.

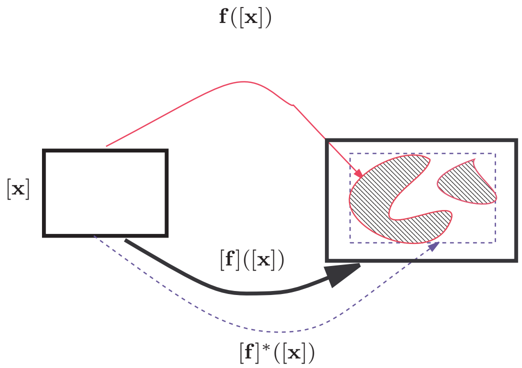

$\mathbf{f}([\mathbf{x}])$
$[\mathbf{x}]$
$[\mathbf{f}]([\mathbf{x}])$
$[\mathbf{f}]^*([\mathbf{x}])$

Fig. 1. Images of a box $[\mathbf{x}]$ by a function $\mathbf{f}$, a pessimistic inclusion function $[\mathbf{f}]$ and the minimal inclusion function $[\mathbf{f}]^*$.

Different algorithms can be used in order to reduce the size of boxes enclosing $\mathbf{f}([\mathbf{x}])$. For the fusion problem considered here, we have chosen to use constraint propagation techniques [29], because of the great redundancy of data and equations.

2) Constraint Satisfaction Problems (CSP): Consider n variables x_i, $i \in \{1, \ldots, n\}$ linked by m relations (or constraints) of the form

$$f_j(x_1, \ldots, x_n) = 0, \quad j = 1, \ldots, m. \tag{2}$$

If we denote by $\mathbf{x}$ the vector $(x_1, x_2, \cdots, x_n)^T$ and by $\mathbf{f}$ the function whose coordinate functions are the f_j: $\mathbf{f} = (f_1, f_2, \cdots, f_m)^T$, we can write these m constraints in vector notation as

$$\mathbf{f}(\mathbf{x}) = \mathbf{0}.$$

Let us assume that vector $\mathbf{x}$ is known to belong to some prior domain $[\mathbf{x}]$, and we want to compute the set of all $\mathbf{x}$ in the prior domain verifying the constraints. This defines a constraint satisfaction problem (CSP), which can be denoted as

$$\mathcal{H} : (\mathbf{f}(\mathbf{x}) = \mathbf{0}, \mathbf{x} \in [\mathbf{x}]). \tag{3}$$

The solution set of $\mathcal{H}$ is defined as:

$$\mathbf{S} = \{\mathbf{x} \in [\mathbf{x}] \mid \mathbf{f}(\mathbf{x}) = \mathbf{0}\}. \tag{4}$$

Note that $\mathbf{S}$ is not necessarily a box. Under the interval framework, solving the CSP implies finding a box $[\mathbf{x}]'$ such that $\mathbf{S} \subseteq [\mathbf{x}]' \subseteq [\mathbf{x}]$. Figure 2 illustrates a simple CSP with two variables and a single constraint.

Contracting $\mathcal{H}$ means replacing $[\mathbf{x}]$ by a smaller domain $[\mathbf{x}]'$ such that $\mathbf{S} \subseteq [\mathbf{x}]' \subseteq [\mathbf{x}]$. A *contractor* for $\mathcal{H}$ is any operator that can be used to contract $\mathcal{H}$.

There are different kinds of methods to develop contractors. Each of these methods may be particularly well suited to certain types

TABLE II
MM METHODS COMPARISON

MM method	Idea, advantages and drawbacks or limitations
Geometrical approaches [9][47][50]	simple search concept, easy to implement generates unstable results in dense urban area lack of map topology information sensitive to map imprecision
Quddus et al. [36]	geometric algorithm, relatively simple to implement similarity in orientation lack of road widths and connectivity poor detection of missing data without adaptation no Multiple Hypothsesis assumption
El Najjar and Bonnifait [17]	multiple-criteria fusion using Belief function theory proximity and similarity in orientation are used lack of road connectivity considers missing data no Multiple Hypothsesis assumption
Syed and Cannon [46]	fuzzy logic theory is used confidence regions are used for selecting the correct road proximity and similarity in orientation are used map error and navigation system error are not considered not easy to implement and relatively complex
Pyo et al. [35]	multiple hypothesis technique is used topology analysis is used for deriving a set of hypothesis road width is not used probability theory is used for managing multiple hypothesis map erro is not considered
BMM	multi-criteria fusion using Belief function theory use of similarity and topology criteria adapted to bounded representation of the state naturally adapted to detecting missing data handles Multiple Hypothsesis case relatively simple to implement use of orientation criterion is possible but needs small adaptation

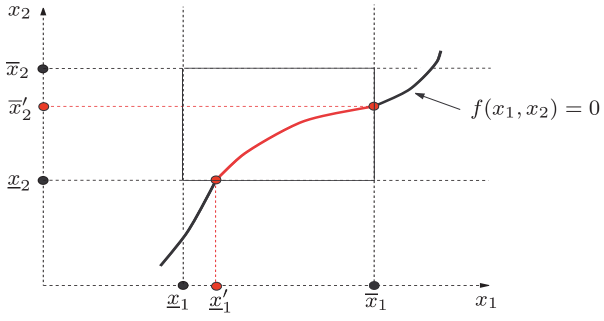


Fig. 2. A CSP in $\mathbb{R}^2$.

of CSP. The method used in this paper is the forward-backward propagation technique, also referred to as the *Waltz contractor* [29].

The Waltz contractor is based on the propagation of *primitive constraints* for real variables. A primitive constraint is a constraint involving a single operator (such as $+, -, *$ or $/$) or a single function (such as $\cos$, $\sin$ or $\sinh$). The method proceeds by contracting $\mathcal{H}$ with respect to each of the primitive constraints until convergence to a minimal domain. The complete description of the Waltz algorithm goes beyond the scope of this paper. It can be found in [29, page

77]. This algorithm was applied to state estimation in [23] and [33].

B. Belief Function Theory

1) Basic Concepts: In recent decades, the theory of belief functions, also known as Dempster-Shafer or Evidence theory, has emerged as a flexible framework for handling imprecise and uncertain information [39][45][51]. Belief functions can be used to replace the probability-based representation of uncertainty adopted in classical MM and MHT methods. The theory of belief functions makes it possible to model various states of knowledge ranging from complete ignorance to probabilistic uncertainty [51]. A belief function may be viewed both as a generalized set [15] and as a generalized probability measure: DS theory thus encompasses set-based as well as probabilistic formalisms.

In this section, we introduce the main concepts of Belief function theory. Let Ω denote a finite set of mutually exclusive and exhaustive hypotheses, called the frame of discernment. A *mass function* m is a mapping from 2^Ω to $[0, 1]$, satisfying:

$$\sum_{A \subseteq \Omega} m(A) = 1.$$

Each quantity $m(A)$ represents a mass of belief that is committed to A, and cannot be assigned to any strict subset of A, based on a given piece of evidence. Every subset A of Ω such that $m(A) > 0$ is called a *focal set* of m. If $m(\emptyset) = 0$ then m is said to be normal.

The *belief function* induced by m is the function $bel : 2^\Omega \longmapsto [0, 1]$ satisfying:

$$bel(A) = \sum_{\emptyset \neq B \subseteq A} m(B) \text{ for all } A \subseteq \Omega. \tag{5}$$

A categorical belief function is a belief function that satisfies $m(A) = 1$ for some $A \subset \Omega$. The belief function on Ω which has $m(\Omega) = 1$ is the vacuous belief function.

The *plausibility function*, denoted pl, quantifies the maximum amount of potential specific support that could be given to $A \subseteq \Omega$. It is obtained by adding all the masses given to focal sets B that satisfy $B \cap A \neq \emptyset$:

$$pl(A) = \sum_{B \cap A \neq \emptyset} m(B) = bel(\Omega) - bel(\overline{A}), \tag{6}$$

where $\overline{A}$ denotes the complement of A.

Assume that a source of information provides a mass function m, and that we have a degree of confidence $\alpha \in [0, 1]$ in the reliability of that source. Then, m can be *discounted* with discount rate $1 - \alpha$, resulting in the following discounted mass function [39]:

$$^{\alpha}m(A) = \begin{cases} \alpha\, m(A) & \text{if } A \subset \Omega, \\ 1 - \alpha(1 - m(\Omega)) & \text{if } A = \Omega. \end{cases}$$

Two mass functions m_1 and m_2 defined on the same frame of discernment Ω and induced by distinct sources of information can be combined by the *conjunctive rule* [44] defined as:

$$(m_1 \mathbin{\bigcirc\mkern-15mu\cap} m_2)(A) = \sum_{B \cap C = A} m_1(B) m_2(C), \quad \forall A \subseteq \Omega. \tag{7}$$

The quantity $(m_1 \mathbin{\bigcirc\mkern-15mu\cap} m_2)(\emptyset)$ is called the *degree of conflict* between the information sources. Under the so-called *open-world assumption* [44], a high degree of conflict may sometimes be interpreted as resulting from the non exhaustivity of the frame of discernment, i.e., the existence of a hypothesis not included in Ω.

Let m be a mass function on Ω after combining all available items of evidence. Assume that we have to select an element of Ω. Several decision rules have been proposed. The most common ones consist in selecting the element with the highest plausibility [5], [12], or the element with the highest *pignistic probability* [45]. Given a mass function m such that $m(\emptyset) < 1$, the pignistic probability function $betp$ is defined as:

$$betp(\omega) = \sum_{\{A \subseteq \Omega | \omega \in A\}} \frac{m(A)}{(1 - m(\emptyset))|A|}, \quad \forall \omega \in \Omega, \tag{8}$$

where $|A|$ denotes the cardinality of A. The pignistic probability function is thus obtained from m by distributing equally each normalized mass $m(A)/(1 - m(\emptyset))$ among the elements of A.

2) Image of a Mass Function by a Multi-valued Mapping : Let us now consider the case where we have two variables X and Y defined on frames of discernment Ω and Θ. Assume that X and Y are linked by a multi-valued mapping $\varphi : \Omega \to 2^\Theta$, such that if $X = \omega$, then we know that $Y \in \varphi(\omega)$. This mapping can be extended to 2^Ω as follows:

$$\varphi(A) = \begin{cases} \bigcup_{\omega \in A} \varphi(\omega), & \text{if } A \subseteq \Omega, A \neq \emptyset, \\ \emptyset & \text{if } A = \emptyset. \end{cases} \tag{9}$$

Let us further assume that we have a mass function m^Ω on Ω representing our state of knowledge about X. A mass function m^Θ on Θ can be built by transferring each mass $m^\Omega(A)$ to $\varphi(A)$ [13]. Formally, m^Θ is then defined as follows:

$$m^\Theta(B) = \sum_{\{A \subseteq \Omega | \varphi(A) = B\}} m^\Omega(A), \quad \forall B \subseteq \Theta.$$

This may also be written

$$m^\Theta(B) = \sum_{A \subseteq \Omega} M_\varphi(B, A) m^\Omega(A), \quad \forall B \subseteq \Theta, \tag{10}$$

with

$$M_\varphi(B,A) = \begin{cases} 1 & \text{if } B = \varphi(A), \\ 0 & \text{otherwise.} \end{cases}$$

IV. Belief Map Matching Method

In this section, we present a belief function-based MM method (BMM) based on bounded error estimation techniques as well as geometrical and topological information. The method builds a mass function on a set of candidate roads (CRs) extracted from an existing local map, which makes it possible to provide a more accurate estimation of the vehicle position and to handle multiple hypotheses as well as missing data.

First, under the interval framework, and using a state space model, sensor data is integrated with GPS measurements via a multi-sensor fusion algorithm in order to compute an estimated position. Then, from a rectangular road representation of a two-dimensional geographical information system (GIS), a set of CRs is determined based on the proximity between the vehicle position and rectangular roads. An estimate of the vehicle position on each CR is then computed. Finally, the CR with the highest pignistic probability is selected. Note that the estimated positions on all CRs are computed, but only the one corresponding to the best CR is considered as the most plausible vehicle position.

In the following, we first present the geometry of the available rectangular road map in Section IV-A. The state space model is introduced in Section IV-B, and the construction of the mass function on the set of CRs is described in Section IV-C. An overview of the BMM method is then presented in Section IV-D, and the integrity of the method is discussed in Section IV-E.

A. Road Map Representation

Digital maps are usually based on a single line road network representation. Each road is characterized by a finite sequence of points and is thus represented by a set of lines linking these points and representing the centerlines of the road. Figure 3 shows an example of a road modeled by four segments connecting five points $A^1, \ldots, A^5$. Points A^1 and A^5 are the end points of the road and are referred to as *nodes*, while A^2, A^3 and A^4 are referred to *shape points*.

In order to take into account road width and geometrical errors of the map, a piecewise rectangular representation of a road can be defined as illustrated in Figure 4.

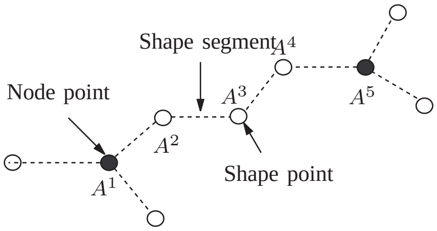


Fig. 3. Road representation in the planar model. Empty circles represent shape points (A^2, A^3, A^4) and filled circles represent node points (A^1, A^5).

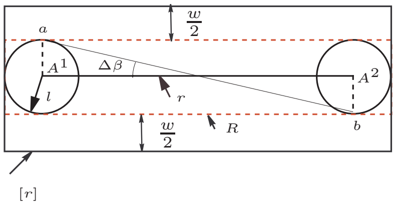


Fig. 4. Rectangular road $[r]$ constructed using map data, geometrical errors and the road width. The road r is represented by two node points (A^1, A^2). Two circles of radius l characterize the positional error. $\Delta\beta$ characterizes the shape error and w is the road width.

By considering the positional error, we assume that the node points of road r can be anywhere within a circle of radius l [34], [37]. The shape error of r is thus represented by an angle $\Delta\beta$. As a result, road r can be considered to lie anywhere within a rectangle R as shown in Figure 4. The rectangular road representation $[r]$ of road r can be constructed by adjusting the width of R using the predefined road width w. Figure 5 shows an example of a road representation where shape points are involved in the construction. In this figure, road r_1 is characterized by a shape point A^2 and two node points A^1 and A^3.

The rectangular representation of roads allows us to take into account several types of errors on digital map data, as it is more exhaustive than other representations used in some existing works [28][18]. However, in some situations (inclined roads), combining a box state with a rectangular road may increase the error on the state estimation result. This error is caused by the *wrapping effect*, which is a well

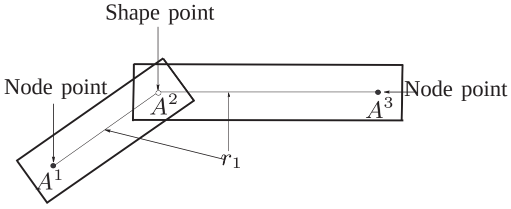


Fig. 5. Piecewise rectangular representation of $r_1 = (A^1, A^2, A^3)$ constructed using geometrical error of the map data and the road width.

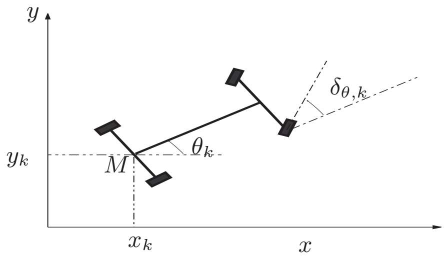


Fig. 6. Definition of the frames.

known issue when using interval methods. It can be reduced using a contractor such as the *Waltz* contractor (cf. Section III-A).

B. Dynamic State Space Model

Consider a car-like vehicle with front-wheel drive. The vehicle position is represented by the Cartesian coordinates (x_k, y_k) of the point M corresponding to the center of the rear axle as shown in Figure 6. The heading angle is denoted by θ_k. The state $\mathbf{x}_k = (x_k, y_k, \theta_k)^T$ is calculated at each time step k using the following discrete representation:

$$\begin{cases} x_{k+1} = & x_k + \delta_{S,k}\cos(\theta_k + \frac{\delta_{\theta,k}}{2}) \\ y_{k+1} = & y_k + \delta_{S,k}\sin(\theta_k + \frac{\delta_{\theta,k}}{2}) \\ \theta_{k+1} = & \theta_k + \delta_{\theta,k}, \end{cases} \tag{11}$$

where $\delta_{S,k}$ is the elementary linear displacement and $\delta_{\theta,k}$ is the measure of the elementary rotation given by an ABS sensor and a gyrometer, respectively. The observation of the position at time step k, $\mathbf{z}_k = (x_{GPS}, y_{GPS})$, is given by a GPS sensor. The longitude and latitude estimated by the GPS are converted to a Cartesian local frame and standard deviations σ_x and σ_y on both coordinates are obtained from the NMEA GPGST (Pseudorange Noise Statistics) sentence[1] [23].

A box is built around $\delta_{S,k}$ and $\delta_{\theta,k}$ using standard deviations σ_s and σ_θ estimated from specific static tests: $[\delta_{S,k}] = [\delta_{S,k} - \kappa \cdot \sigma_s, \delta_{S,k} + \kappa \cdot \sigma_s]$ and $[\delta_{\theta,k}] = [\delta_{\theta,k} - \kappa \cdot \sigma_\theta, \delta_{\theta,k} + \kappa \cdot \sigma_\theta]$, where κ is a constant. In the same way, we build a box around GPS measurement $\mathbf{z}_k$: $[\mathbf{z}_k] = ([x_{k,GPS}], [y_{k,GPS}])^T$ where $[x_{k,GPS}] = [x_{k,GPS} - \kappa \cdot \sigma_x, x_{k,GPS} + \kappa \cdot \sigma_x]$ and $[y_{k+1,GPS}] = [y_{k,GPS} - \kappa \cdot \sigma_y, y_{k+1,GPS} + \kappa \cdot \sigma_y]$.

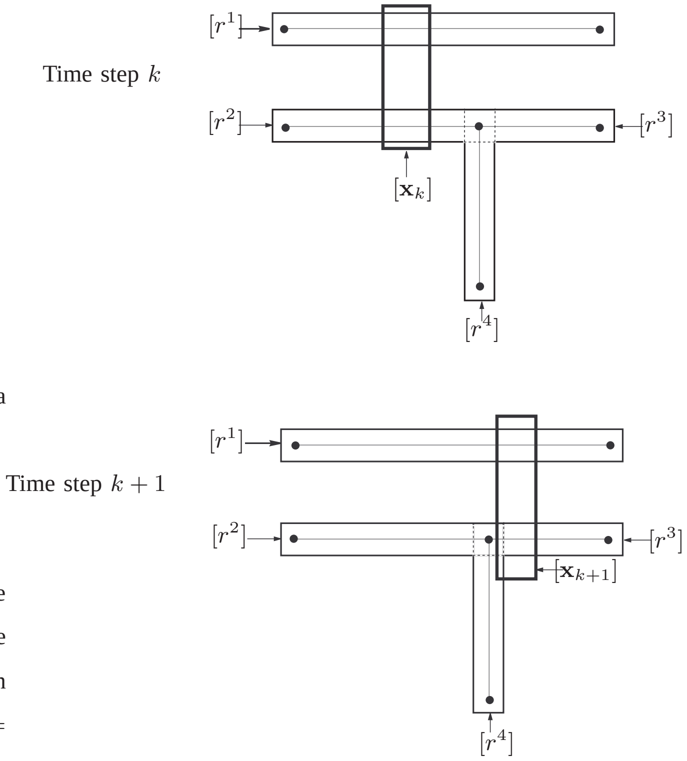


Fig. 7. State box and rectangular roads at time steps k (top) and $k+1$ (bottom).

In the simulations presented below, we have followed common practice in state bounding estimation [1][22] and have chosen $\kappa = 3$. This is justified by the well known fact that, under Gaussian assumptions, the corresponding interval contains the true value of the quantity of interest with 99.87% probability. (From Tchebychev's inequality, this probability cannot be smaller than $8/9$, whatever the error distribution, if the measurement is unbiased). Experimental results have shown that a smaller value such as $\kappa = 2$ might still provide acceptable results. However, the state bounding approach relies on the concept of guaranteed estimation; consequently, the method must be provided with intervals containing unknown quantities with probability close to one.

Note that the value of κ has a critical influence on the precision and reliability of the BMM method. If κ is overestimated, pessimistic state boxes are produced, resulting in imprecise state estimates. Conversely, if κ is underestimated, inconsistencies between measurements, state predictions and map data may occur.

C. Mass Function Construction

Given a state box $[\mathbf{x}_k]$ and a rectangular road map, a set R_k of CRs can be selected in such a way that any rectangular road $[r]$ in

[1] GPS receiver communication is defined within the specification given by the National Marine Electronics Association (NMEA). This specification was first set up in order to define the interface between various pieces of marine electronic equipment. Most computer programs that provide real time position information understand and expect data to be in the NMEA format.

R_k satisfies: $[\mathbf{x}_k] \cap [r] \neq \emptyset$. The associated mass function at time step k, m^{R_k}, can be computed using topology and similarity criteria. This is done as follows:

- Topology criterion: Using the topology of the map and mass function m^{R_k}, a mass function $m_1^{R_{k+1}}$ on R_{k+1} can be computed as shown by the following example.

  Consider the case of Figure 7, where road r^2 is linked to roads r^3 and r^4 and road r^1 is parallel to r^2 and r^3. The vehicle positions at time steps k and $k+1$ are represented by two solid rectangles $[\mathbf{x}_k]$ and $[\mathbf{x}_{k+1}]$. Based on $[\mathbf{x}_k]$, the set of CRs at time step k is $R_k = \{r^1, r^2\}$. The available mass function at time step k is m^{R_k} with focal sets $\emptyset, \{r^1\}, \{r^2\}, \{r^1, r^2\}$. Note that $m^{R_k}(\{r^i, r^j\})$ represents the mass of belief assigned to the hypothesis that the vehicle is on road r^i or r^j at time step k, and $m^{R_k}(\emptyset)$ represents the mass of belief assigned to the hypothesis that the vehicle is moving on a road that is not included in the map database at time step k.

  Based on $[\mathbf{x}_{k+1}]$, four possible positions of the vehicle on r^1, r^2, r^3 or r^4 may occur and thus $R_{k+1} = \{r^1, r^2, r^3, r^4\}$. Consequently, from the fact that r^2 is linked to $\{r^3, r^4\}$, and given that r^1 has no intersections with any other roads, the locations of the vehicle at time steps k and $k+1$ are linked by the following multi-valued function $\varphi : R_k \to 2^{R_{k+1}}$:

$$\varphi(r^1) \quad = \quad \{r^1\}, \tag{12}$$

$$\varphi(r^2) \quad = \quad \{r^2, r^3, r^4\}, \tag{13}$$

  where $\varphi(r)$ denotes the possible positions at time $k+1$, given that the vehicle was on road r at time k. As explained in Section III-B2, φ can be extended to 2^{R_k} using (9), and a mass function $m_1^{R_{k+1}}$ can be built on R_{k+1} using (10). Here, the computation of $m_1^{R_{k+1}}$ from m^{R_k} can be detailed as follows:

  - $m^{R_k}(\emptyset)$ is transferred to $m_1^{R_{k+1}}(\emptyset)$;
  - $m^{R_k}(\{r^1\})$ is transferred to $m_1^{R_{k+1}}(\{r^1\})$;
  - $m^{R_k}(\{r^2\})$ is transferred to $m_1^{R_{k+1}}(\{r^2, r^3, r^4\})$ as r^2 is linked to r^3 and r^4;
  - $m^{R_k}(\{r^1, r^2\})$ is transferred to $m_1^{R_{k+1}}(\{r^1, r^2, r^3, r^4\})$.

  In general, matrix M_φ (hereafter referred to as the *transition matrix*) to be used in (10) is computed from the topology of the road network, and we have

$$m_1^{R_{k+1}}(B) = \sum_{A \subseteq R_k} M_\varphi(B, A) m^{R_k}(A), \quad \forall B \subseteq R_{k+1}. \tag{14}$$

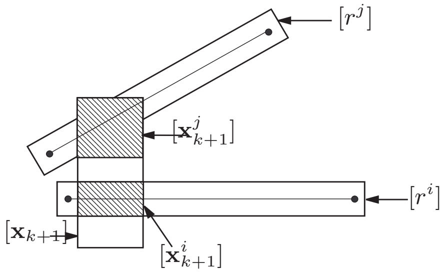


Fig. 8. State update using $[\mathbf{x}_{k+1}]$ and two rectangular roads $[r^i]$ and $[r^j]$.

- Similarity criterion: Using a measure of similarity between the rectangular roads in R_{k+1} and the state box $[\mathbf{x}_{k+1}]$, a mass function $m_2^{R_{k+1}}$ on R_{k+1} is calculated. This similarity is characterized by the area of the intersection between $[\mathbf{x}_{k+1}]$ and the rectangular roads. For geometrical convenience it is calculated as the area of the minimal box containing this intersection. Let

$$L^i = \frac{|[\mathbf{x}^i_{k+1}]|}{|[\mathbf{x}_{k+1}]|},$$

  where $[\mathbf{x}^i_{k+1}]$ is the minimal box containing the intersection between the rectangular road $[r^i]$ and $[\mathbf{x}_{k+1}]$ as shown in Figure 8. L^i can be seen as a geometrical likelihood of the road given a state box $[\mathbf{x}_{k+1}]$. Using L^i, a mass function m_i can be computed as follows [5]:

$$\begin{cases} m_i(\{r^i\}) = & 0 \\ m_i(\overline{\{r^i\}}) = & \alpha_i(1 - L^i) \\ m_i(R_{k+1}) = & 1 - \alpha_i(1 - L^i) \end{cases} \tag{15}$$

  where $\overline{\{r^i\}}$ is the complement of $\{r^i\}$ in R_{k+1} and α_i is a coefficient associated with road r^i. The mass function $m_2^{R_{k+1}}$ is the combination of all m_i using the conjunctive rule of combination (7):

$$m_2^{R_{k+1}} = \textcircled{\cap}_i m_i. \tag{16}$$

The mass functions $m_1^{R_{k+1}}$ and $m_2^{R_{k+1}}$ are then combined in order to compute $m^{R_{k+1}}$ as

$$m^{R_{k+1}} = m_1^{R_{k+1}} \textcircled{\cap} m_2^{R_{k+1}}. \tag{17}$$

D. Overview of the BMM Method

1) Initialization: At time step $k = 0$, an initial state box is constructed using the GPS measurement and the standard deviations σ_x and σ_y estimated in real time by the GPS receiver: $[x_0] = [x_{0,GPS} - 3 \cdot \sigma_x, x_{0,GPS} + 3 \cdot \sigma_x]$ and $[y_0] = [y_{0,GPS} - 3 \cdot \sigma_y, y_{0,GPS} + 3 \cdot \sigma_y]$. Note that the heading angle θ is not directly observed, and is initialized as $[\theta_0] = [0, +2\pi]$. From $[\mathbf{x}_0]$ and the rectangular road map, a set R_0 of CRs is selected as explained in Section IV-C. As there is no prior information on the vehicle position at time step $k = 0$, $m_1^{R_0}$ should be initialized as a vacuous mass function on R_0. The mass function $m_2^{R_0}$ is calculated using (15) and (16). The final mass function m^{R_0} is thus the result of the combination of $m_1^{R_0}$ and $m_2^{R_0}$ according to (17). As the vehicle should be on a road, $[\mathbf{x}_0]$ is replaced by $\{[\mathbf{x}_0^i]\}_{i=1}^{n_{R_0}}$ as shown in Figure 8.

2) Prediction: In this step, state boxes $\{[\mathbf{x}_k^i]\}_{i=1}^{n_{R_k}}$ associated with all CRs are updated using $[\delta_{S,k}]$, $[\delta_{\theta,k}]$ together with the evolution equation (11) via the application of the interval tools recalled in Section III-A [23], [29]. Note that, as inclusion functions are used, we may obtain *suboptimal* predicted state boxes $\{[\mathbf{x}_{k+1}^i]\}_{i=1}^{n_{R_k}}$ at time step $k+1$.

3) GPS correction: The GPS measurement box $[\mathbf{z}_{k+1}]$ is used in order to adjust state boxes. The intersection between a state box $([x_{k+1}^i], [y_{k+1}^i])^T$ and the GPS box $[\mathbf{z}_{k+1}]$: $[I^i] = [\mathbf{x}_{k+1}^i] \cap [\mathbf{z}_{k+1}]$, characterizes the proximity between the prediction and the measurement. This intersection is used to contract $[\mathbf{x}_{k+1}^i]$ using the Waltz algorithm (see Section III-A) according to the constraints of system (11) [1] [29]. Note that CRs associated with the predicted state boxes that have no intersection with $[\mathbf{z}_{k+1}]$ are eliminated from R_{k+1}.

4) GIS correction: After the GPS correction, the piecewise rectangular representation of CRs is used to adjust the state boxes and to compute a state estimate of the vehicle. Figure 9 shows an example where, starting from road r^1, there is no possible transition to another road, whereas there is a possibility of transition from r^2 to r^3 or r^4.

Regarding junction situations, two cases should be considered:

- If the distance between the center of $[\mathbf{x}_k^i]$ and any node or shape point of road r^i is less than the elementary displacement $\delta_{S,k}$ given by the rear wheel ABS sensors, then it is possible that the vehicle is leaving road r^i. For this reason, at time step $k+1$, the set R_k of CRs must be replaced by a set R_{k+1}. This is done as follows. Let S^i be the set of all roads directly linked to r^i ($r^i \in S^i$) and that have an intersection with $[\mathbf{x}_k^i]$. The set R_{k+1} is then updated as: R_{k+1}=$R_k \cup S^i$.
  Using a transition matrix M_φ calculated from the topology of the map, the mass function $m_1^{R_{k+1}}$ is then calculated from m^{R_k} from (14).
- If the distance between the center of state box $[\mathbf{x}_k^i]$ and each node and shape point of road r^i is greater than $\delta_{S,k}$, then $R_{k+1} = R_k$.

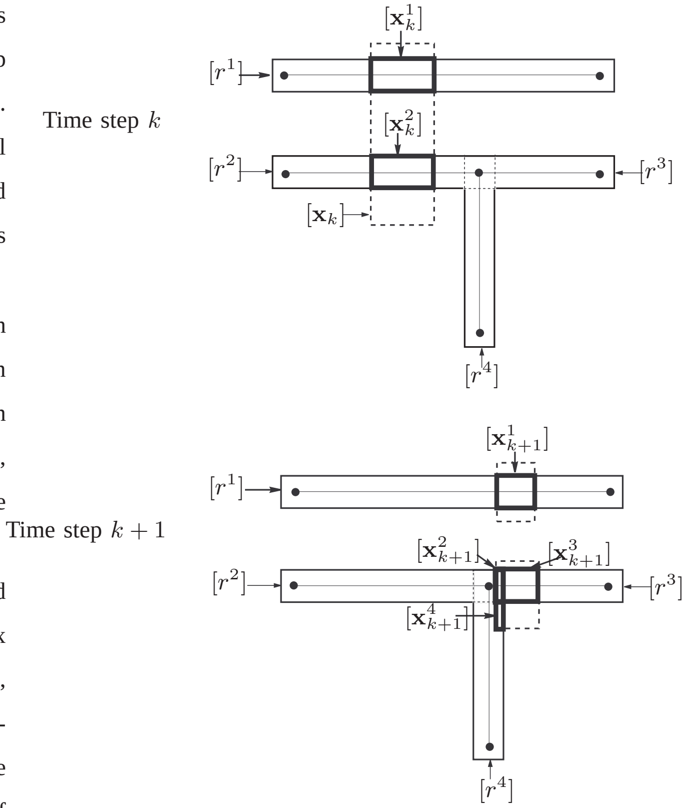


Fig. 9. Possible scenarios when managing multiple hypotheses due to junctions. The bold black boxes represent the state after GIS correction.

The mass function $m_2^{R_{k+1}}$ resulting from the similarity criterion is computed using (15) and (16). By combining $m_1^{R_{k+1}}$ and $m_2^{R_{k+1}}$ using (17), the final mass function $m^{R_{k+1}}$ is then obtained.

Note that, if there is no intersection between state boxes and any road in R_{k+1}, then the resulting state boxes $\{[\mathbf{x}_{k+1}^i]\}_{i=1}^{n_{R_k}}$ should be retained, as the vehicle may be moving on a road that is not included in the map. In this case, we update the set of CRs as follows: $R_{k+1} = R_k \cup T$, where T includes all roads linked to any $r^i \in R_k$. This rule is introduced as an attempt to match the estimated position of the vehicle on a road in case of failure of the algorithm. If this solution is not adequate, an initialization of the algorithm is needed. In the other case, i.e., if at least one road r^j is matched, each available box $[\mathbf{x}_{k+1}^i]$ is replaced by the minimal box $[\mathbf{x}_{k+1}^j]$ containing the intersection between $[\mathbf{x}_{k+1}^i]$ and the rectangular road $[r^j]$.

5) Overall Estimation: In the final step of the algorithm, the CR with the highest pignistic probability (8) is selected, and the state box intersecting the selected road is chosen. In the case where several boxex intersect the selected road, the state box is taken as the minimal box enclosing all boxes associated with the road. The point estimate of the vehicle position is defined as the center of the final state box.

The confidence in the estimated coordinates x and y is then defined as the width of the box along the corresponding dimension.

The BMM algorithm is summarized in Appendix A. The integrity issue is briefly discussed in the following subsection.

E. Integrity of the BMM Method

The integrity of a map-matching algorithm is the measure of the confidence that can be placed in the correctness of the positions delivered by the system [37]. It is related to the ability of the system to correctly identify a link and to accurately determine the vehicle location on the link. In [37], the authors propose an integrity measure based on three criteria:

- Distance residuals: This criterion quantifies the proximity between the vehicle position fix P and the corresponding map-matched position M on a link. Let $D(P, M)$ be the distance between P and M and R be a constant computed based on the noise of the navigation system and the road width. The integrity of the map is high if $D(P, M) - R$ is less than or close to zero.
- Heading residuals: This criterion quantifies the proximity between the vehicle and the road heading. Let θ be the vehicle heading, β the road heading and HE a constant computed based on the map error and the navigation system error. The integrity of the map is high if $|\theta - \beta| - HE$ is less than or close to zero.
- Confidence of the map matched position: This criterion is based on the proximity between the map matched position M and the real vehicle position V. Let $D(M, V)$ be the distance between M and V. The confidence of the map matched position is high if $D(M, V) \leq r_f$ where r_f is a constant computed from the map error, sensor error and road width.

The BMM method has the advantage of naturally meeting the integrity criteria mentioned above. The map matched position of the vehicle is included in the intersection of the bounded estimation box and the rectangular representation of the roads. As the map error and road width are taken into account when constructing the rectangular roads of the map, and the noise sensors are integrated in the bounded estimation of the vehicle position, it is clear that $D(P, M) - R$ and $|\theta - \beta| - HE$ are usually less than or close to zero. Since the real vehicle position is included in the box provided by the GPS/DR system with probability close to one, $D(V, M)$ is usually small and is less than or equal to $\frac{w}{2} + l$, where w is the road width and l represents the geometrical error of the map. We may therefore conclude that the BMM method naturally complies with the integrity requirements proposed in [37].

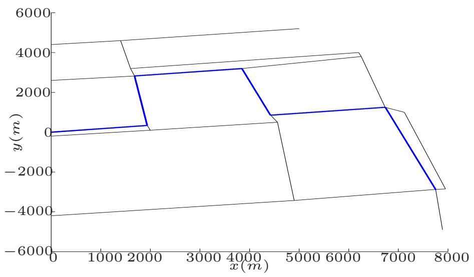


Fig. 10. Simulated road map and vehicle trajectory (blue bold lines).

V. Experiments

In this section, we first illustrate the behavior of the BMM method using simulated data and show comparative results in Subsection V-A. Results with real data are then presented in Subsection V-B.

A. Simulated Results

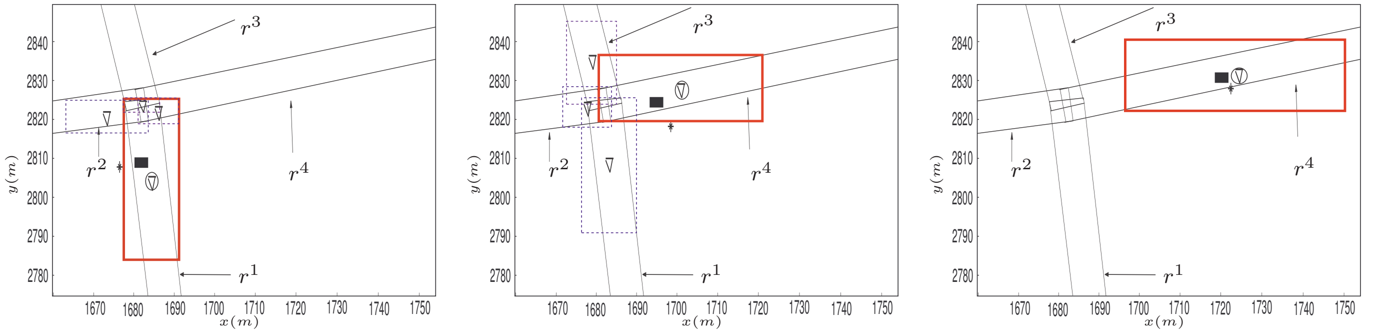
The performances of the BMM algorithm were studied using simulation data. The vehicle position, heading, elementary displacement and elementary rotation were generated using the Matlab Simulink toolbox. Uniform GPS measurement noise was used, with bounds $[-7m, +7m]$ and $[-9m, +9m]$ on x and y respectively. The noise in the input data (elementary movement and rotation) was also assumed to be uniform with on the interval $[-0.25m, +0.25m]$ and $[-0.002\,deg, +0.002\,deg]$. The positional error l and the road width w were assumed to be equal to 1 and 6 meters, respectively. This leads to a rectangular road width equal to 8 meters ($w + 2 * l$). Figure 10 shows the simulated map and the vehicle trajectory. All distances in the following figures are in meters.

Figure 11 shows how several hypotheses can be managed with the BMM method when approaching a junction.

Figure 11 illustrates the management of multiple hypotheses by the BMM method after time step $k = 182$. As can be seen, at time steps $k = 182$ and $k = 183$, there are four CRs $R_k = \{r^1, r^2, r^3, r^4\}$. At time step $k = 184$, there is only one CR left $r_k = \{r^4\}$, as the other three $\{r^1, r^2, r^3\}$ are eliminated by the similarity criterion.

1) Performance comparison: Our method was compared with two recent MM algorithms proposed by Quddus et al. [36] and by El Najjar and Bonnifait [17], [18]. We first give a short description of these two algorithms:

- The algorithm developed by Quddus et al. [36] is a relatively simple geometric algorithm. The MM matching process is initiated with point-to-point matching to identify a link among



$k = 182$ $k = 183$ $k = 184$

Fig. 11. Simulation results at a junction starting at $k = 182$. Rectangular roads are represented by solid rectangles. The real and GPS positions are represented by (■) and ($*$) points, respectively. Dashed blue rectangles represent the box state associated to CRs where the associated estimated positions are plotted by (∇) points. The state box corresponding to the selected road is plotted as a bold red rectangle and the overall estimated positions are represented by (o) points.

the links connected with the closest node to the position fix. Based on various similarity criteria between the derived position fixes and the network topology, a weighting system is used to select the correct link. The criteria used in the algorithm are the similarity in orientation, proximity of the point to the link and the position of the fix relative to the link.

- The advanced MM algorithm introduced by El Najjar and Bonnifait in [17], [18] is based on Kalman filtering and belief functions. The EKF first combines the GPS and ABS sensor measurements to produce an approximation of the vehicle position, which is used to select the most likely road segment in the database. The selection strategy uses several criteria based on distance, direction and velocity measurements, each expressed as a mass function and pooled using the conjunctive rule of combination. A new observation is then built using the selected segment, and the estimated position is adjusted in a second Kalman filter stage.

TABLE III

MEAN SQUARED ERRORS ON x AND y FOR THE GPS, THE TWO REFERENCE METHODS AND THE BMM ALGORITHM.

	GPS	[36]	[17]	BMM
MSE on $x(m^2)$	25.3	14.1	12.0	10.7
MSE on $y(m^2)$	27.8	17.6	14.6	12.3
one step running time (ms)	-	81	93	98

These three methods were applied to the simulated map in Figure 10. Mean squared errors (MSE) on x and y are reported in Table III (results with the GPS alone are also included for comparison). As can be seen, the advanced method developed by El Najjar and Bonnifait outperforms the simple geometric method developed by Quddus et al., and the BMM algorithm brings further improvement. We remark that the latter two methods use the theory of belief functions to combine the road selection criteria. The main difference between the two methods resides in the fact that El Najjar and Bonnifait use a Bayesian approach for state estimation, whereas the BMM method uses a bounded error approach for both state estimation and road representation. Table III also gives the mean of the execution time of one step for each algorithm using a 3GHz Pentium 4 and a Matlab implementation. It is clear that the BMM method satisfies real time constraints despite the use of interval arithmetic programs under Matlab and without code optimization. We note that the other methods have a small advantage over the BMM method inasmuch as the BMM method handles the case of multiple-hypothesis and also in general takes into account more attributes of the map database.

2) Result with Missing Map Data: In this section, we demonstrate the behavior of the BMM method when some roads are missing from the map. Figure 12 illustrates such a situation. As can be seen in Figure 12-b, after time step $k = 66$ the vehicle leaves road r^4 and proceeds to travel along a road that is not included in the map. At time step $k = 88$, the vehicle is once again travelling on an existing road, r^5.

Figure 13-a shows the zoomed result at time steps $k = 66$ and $k = 67$. As can be seen, at time step $k = 66$ there is only one CR and thus $R_k = \{r^4\}$ and $m^{R_k}(\{r^4\}) = 1$. At time step $k = 67$ there is no intersection between the state box and the CR. In this case, it is assumed that the vehicle is travelling on a road

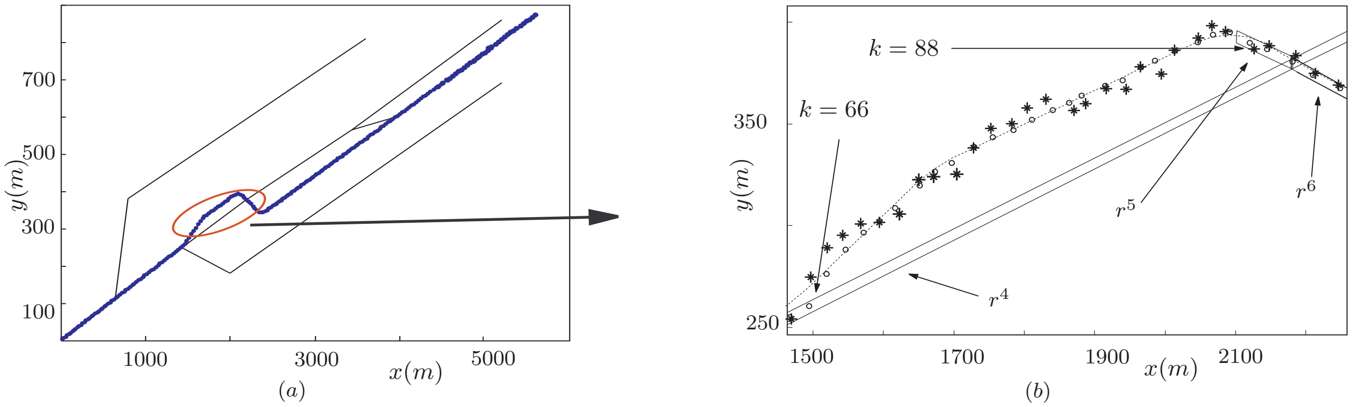


Fig. 12. Simulated map and trajectory in the case of missing map data. Figure (a) shows the vehicle trajectory represented by bold blue lines. Figure (b) shows the vehicle trajectory (represented by dashed line) on missing road between time steps $k = 66$ and $k = 88$.

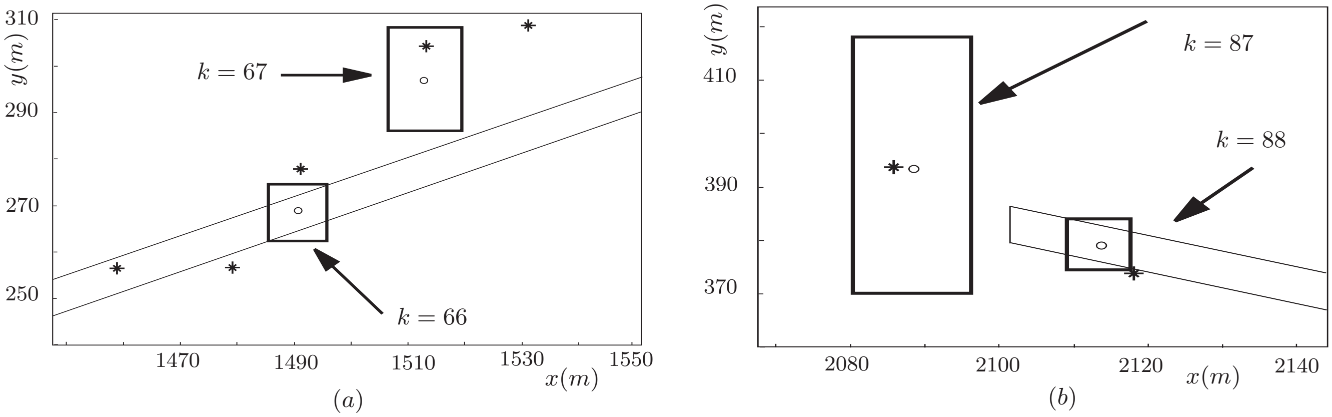


Fig. 13. Missing map data case given in the Figure 12-b. GPS positions are represented by black $(*)$ points and estimated positions are represented by (o) points. Roads and estimated state boxes are represented by solid and bold rectangles, respectively. Figure (a) shows time step $k = 66$ where $R_k = \{r^4\}$ and $m^{R_k}(\{r^4\}) = 1$, and time step $k = 67$ where $R_k = \{r^4, r^5, r^6\}$ and $m^{R_k}(\phi) = 1$. Figure (b) shows time step $k = 87$ where $R_k = \{r^4, r^5, r^6\}$ and $m^{R_k}(\phi) = 1$, and time step $k = 88$ where $R_k = \{r^5\}$ and $m^{R_k}(\{r^5\}) = 1$.

that is not included in the database. Consequently, the set of CRs is updated to $R_k = \{r^4, r^5, r^6\}$ as explained in Section IV-D4, and $m^{R_k}(\emptyset) = 1$. At time step $k = 88$ (Figure 13-b), there is an intersection between the rectangular representation of road r^5 and the state box. Consequently, the vehicle is considered to be travelling on r^5 and $m^{R_k}(\{r^5\}) = 1$. Figure 14 shows the results of applying the BMM and reference methods to the trajectory depicted in Figure 12. Globally, both methods perform well as they detect the incompleteness of the map for the most part of the trajectory. However, the BMM algorithm detects earlier that the vehicle has left the main road ($k = 67, 68$), and the reference method tends to select the nearest road on the map too early at time $k = 87$.

B. Results on Real Data

In this section, results with the BMM method applied to real data are reported. The test trajectory was carried out in Compiègne, France with the experimental vehicle of the Heudiasyc laboratory[2]. The measurement of the position (x_{GPS}, y_{GPS}) was provided by a GPS receiver. The elementary rotation and displacement between two samples were obtained with good precision using a fiber optic gyrometer and two rear wheel ABS sensors. In this application, the positional error l was assumed to be equal to one meter, and the road width w was equal to 6 meters. The map is shown in Figure 15.

Figure 16-a displays the BMM estimated positions near a road junction (time step k_1). Figures 16-b and c show the estimated positions corresponding to all CRs at time steps k_1 and $k_1 + 1$.

[2] We thank Philippe Bonnifait and his group for providing and allowing us to use these data.

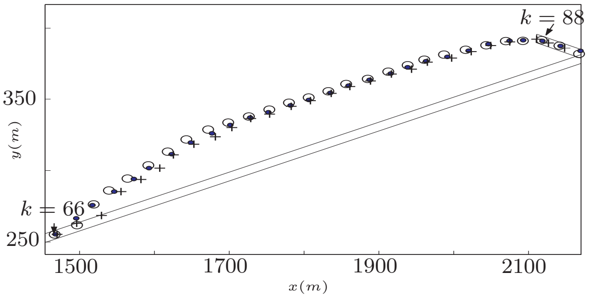


Fig. 14. Comparison between the BMM algorithm and the method developed by El Najjar and Bonnifait (reference method) in the case of missing map data. The true positions of the vehicle are plotted by $(.)$ points. Estimated positions computed using the BMM and reference methods are plotted as (o) points and $(+)$ points, respectively.

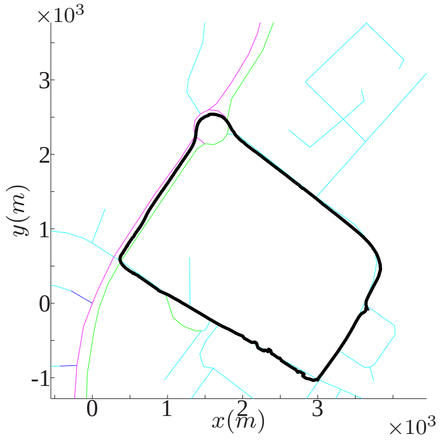


Fig. 15. Test trajectory and digital map. Bold lines represent the vehicle position estimated by the GPS.

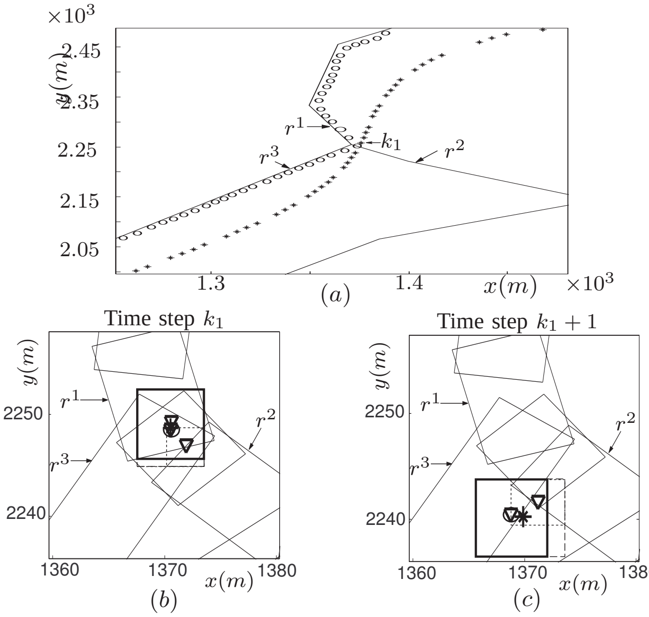


Fig. 16. Experimental results at a junction. BMM results and GPS positions are plotted as (o) and $(*)$ points, respectively (a). In sub-figures (b) and (c), the estimated positions on CRs, overall estimated positions and GPS measurements are represented by (∇) points, (o) points and $(*)$ points, respectively. The dashed and bold rectangles represent the box states of CRs and the box states corresponding to the selected road, respectively.

TABLE IV
COMPARISON OF BMM, QUDDUS METHOD AND REFERENCE METHODS

	[36]	[17]	BMM
Correct link Identification (%)	89.2	96.8	99.2

As can be seen, at time step k_1, there are three CRs and the BMM method provides an estimate of the position on each CR. At time step $k_1 + 1$, only two CRs remain (see Figure 16-c), as the third one has been eliminated by the BMM algorithm.

As the exact trajectory of the vehicle is not known in this application, positioning errors cannot be computed like they were in the simulation Section. However, the performances of the BMM method can be assessed quantitatively by computing the fraction of time the correct link was selected. The correct link at each time step is known here thanks to a camera that was fixed on the vehicle during the experiment. A data player makes it possible to visualize the road scenes together with the sensor data and to identify the true trajectory on the map. Table IV reports the fractions of time (out of 1500 samples) when the correct link was identified by the BMM algorithms. Results obtained with the methods developed by Quddus et al. [36] and El Najjar and Bonnifait [17], [18] are also shown for comparison. As can be seen, the advanced method developed El Najjar and Bonnifait significantly outperforms the geometric reference method, while the BMM algorithm brings further improvement. The scale of the digital map is of no importance in the evaluation of the algorithm, since only the percentage of detected road is given.

VI. CONCLUSION

In this paper, a new method for map matching and state estimation has been presented. This method combines the outputs from existing bounded error estimation techniques with piecewise rectangular roads selected using evidential reasoning. The basic idea of this work is the selection of a set of CRs at each time step using the topology of the map and a similarity criterion. Then, a mass function on the set of CRs is calculated using a multiple criteria fusion algorithm. In this method two criteria are used: map topology and similarity. After selecting CRs and computing the associated mass function, an overall estimate of the vehicle position is calculated using a decision rule from belief function theory. This method enables us to handle multiple hypotheses efficiently at road junctions, and to cope with missing map data. In addition, its implementation is quite simple as it is based on geometrical properties of boxes and rectangular roads. Results on simulated and real data have demonstrated the ability of the method to handle ambiguous situations (such as junctions or close parallel roads) and to compute an accurate and reliable estimation of the vehicle position.

APPENDIX A
SUMMARY OF THE BMM ALGORITHM

1) Initialization
    a) Set $k = 0$ and create a state box $[\mathbf{x}_0]$ using the GPS measurement
    b) For each road r^i included in the local map, construct the associated rectangular road $[r^i]$
    c) From all rectangular roads $[r^i]$ and $[\mathbf{x}_k]$, construct a set R_k of CRs such that $[r^i] \cap [\mathbf{x}_k] \neq \emptyset$. Let n_{R_k} be the number of roads included in R_k.
    d) State update: $[\mathbf{x}_k^i] = [\mathbf{x}_k] \cap [r^i]$, $i = 1, \ldots, n_{R_k}$.
    e) Construct a mass function $m_2^{R_k}$ using (15) and (16) on R_k. Set $m^{R_k} = m_2^{R_k}$
2) Set $R_{k+1} = R_k$ and $m_1^{R_{k+1}} = m^{R_k}$
3) For $i = 1 : n_{R_k}$
4) Prediction:
    - Input boxes: $[\delta_{S,k}] = [\delta_{S,k} - 3 \cdot \sigma_s, \delta_{S,k} - 3 \cdot \sigma_s]$ and $[\delta_{\theta,k}] = [\delta_{\theta,k} - 3 \cdot \sigma_\theta, \delta_{\theta,k} - 3 \cdot \sigma_\theta]$
    - Calculate $[\mathbf{x}_{k+1}^i]$ using $[\delta_{S,k}]$, $[\delta_{\theta,k}]$ and (11)
5) GPS correction:
    a) Using the GPS data, build a measurement box: $[\mathbf{z}_{k+1}] = ([x_{k+1,GPS}], [y_{k+1,GPS}])'$, $[x_{k+1,GPS}] = [x_{k+1,GPS} - 3 \cdot \sigma_x, x_{k+1,GPS} + 3 \cdot \sigma_x]$ and $[y_{k+1,GPS}] = [y_{k+1,GPS} - 3 \cdot \sigma_y, y_{k+1,GPS} + 3 \cdot \sigma_y]$
    b) The innovation is given by: $[I^i] = [\mathbf{x}_{k+1}^i] \cap [\mathbf{z}_{k+1}]$
    c) **IF** $[I^i]$ is not empty then
        - Contract $[\mathbf{x}_{k+1}^i]$ using $[I^i]$ and by applying the Waltz algorithm according to system (11)
    d) **ELSE** $[\mathbf{x}_{k+1}^i]$ is eliminated
6) GIS correction:
    a) Update R_{k+1} and $m_1^{R_{k+1}}$:
        - **IF** the distance between the center of $[\mathbf{x}_k^i]$ and a node or a shape point of road r^i is less than $\delta_{S,k}$, then:
            – Let $S(r^i)$ be the set of all roads directly linked to r^i including r^i. Let n_s be the number of roads included in $S(r^i)$. Initialize $S^i = \emptyset$
            – For $j = 1 : n_s$
            – if $[r^j] \cap [\mathbf{x}_k^i]$ is not empty then
                ∗ $S^i = S^i \cup \{r^j\}$
                ∗ $[\mathbf{x}_{k+1}^j] = [\mathbf{x}_{k+1}^i] \cap [r^j]$
            – ENDFOR j
            – $R_{k+1} = R_{k+1} \cup S^i$ and $m_1^{R_{k+1}} = M.m_1^{R_{k+1}}$, where M is the transition matrix computed in such way that $m_1^{R_{k+1}}(\{r^i\})$ is transferred to $m_1^{R_{k+1}}(S^i)$
        - **ELSE** R_{k+1} and $m_1^{R_{k+1}}$ remain unchanged and: $[\mathbf{x}_{k+1}^i] = [\mathbf{x}_{k+1}^i] \cap [r^i]$
        - **ENDIF**
        - **IF** $\{[\mathbf{x}_k^i] \cap [r^i] = \emptyset\}_{i=1}^{n_{R_k}}$ then
            – $\{[\mathbf{x}_{k+1}^i]\}_{i=1}^{n_{R_k}}$ are kept
            – $R_{k+1} = R_k \cup T$, T includes all roads linked to r^i in R_k
            – If this solution is not adequate, then an initialization step is needed
        - **ENDIF**
    b) Construct a mass function $m_2^{R_{k+1}}$ on R_{k+1} using (15) and (16)
7) ENDFOR i
8) Compute $m^{R_{k+1}} = m_1^{R_{k+1}} \circledcirc m_2^{R_{k+1}}$
9) Overall estimation:
    a) Select the road with the highest pignistic probability computed from $m^{R_{k+1}}$
    b) The state box is the smallest box enclosing all boxes associated to the best road
    c) The state estimate is the center of the state box
10) $k = k + 1$. Go to 2 until $k = k_{end}$

REFERENCES

[1] F. Abdallah, A. Gning and Ph. Bonnifait. *Box Particle Filtering for non Linear State Estimation using Interval Analysis*. Automatica, volume 44, pp. 807-815, 2008.

[2] Agamennoni, G.; Nieto, J.I.; Nebot, E.M.. *Robust Inference of Principal Road Paths for Intelligent Transportation Systems*. IEEE Transactions on Intelligent Transportation Systems, vol. 12, Issue:1, pp. 298-308, 2011.

[3] Alvarez, J.M.A.; Lopez, A.M.. *Road Detection Based on Illuminant Invariance*. IEEE Transactions on Intelligent Transportation Systems, vol. 12, Issue:1, pp. 184-193, 2011.

[4] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. *A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking*. IEEE Trans. on Signal Processing, Vol.50, No.2, pp 174-188, 2002.

[5] A. Appriou. Uncertain data aggregation in classification and tracking processes. In B. Bouchon-Meunier, editor, *Aggregation and Fusion of imperfect information*, pages 231–260. Physica-Verlag, Heidelberg, 1998.

[6] A. Asadian, B. Moshiri and A.K. Sedigh. *A Novel Data Fusion Approach in an Integrated GPS/INS System Using Adaptive Fuzzy Particle Filter*. 5th International Conference on Technology and Automation (ICTA), Sponsored by IEEE and EURASIP, 15-16 October 2005, Thessaloniki, Greece.

[7] I. Ashokaraj, A. Tsourdos, P. Silson and B. White. *Sensor Based Robot Localisation and Navigation: Using Interval Analysis and Extended Kalman Filter*. 5th Asian Control Conference (ASCC),2004.

[8] Y. Bar-Shalom and W. D. Blair. *Multitarget-Multisensor Tracking: Applications and Advances*, vol. III. Artech House, 2000.

[9] D. Bernstein, A. Kornhauser. *An introduction to map matching for personal navigation assistants*. New-Jersey TIDE Center, 1996.

[10] W.S. Chen. *Bayesian estimation by sequential Monte Carlo sampling*. Dissertation, The Ohio State University, 2004.

[11] W. Chen, Z. Li, M. Yu and Y. Chen. *An integrated map-match algorithm with position feedback and shape-based mismatch detection and correction*.Journal of Intelligent Transportation Systems. 12(4): 168-175, 2008.

[12] T. Denœux. *Analysis of evidence-theoretic decision rules for pattern classification*. Pattern Recognition, 30(7):1095-1107, 1997.

[13] T. Denœux. *Inner and outer approximation of belief structures using a hierarchical clustering approach*. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 9(4):437–460, 2001.

[14] Drawil, N.M.; Basir, O.. *Intervehicle-Communication-Assisted Localization*. IEEE Transactions on Intelligent Transportation Systems, vol. 11, Issue:3, pp. 678-691, 2010.

[15] D. Dubois and H. Prade. *A set-theoretic view of belief functions: logical operations and approximations by fuzzy sets*. International Journal of General Systems, 12(3):193–226, 1986.

[16] M. E. El Najjar and PH. Bonnifait. *Multi-criteria fusion for the selection of roads of an accurate map*. 15^{th} IFAC World Congress, Barcelona, 2002.

[17] M. E. El Najjar and PH. Bonnifait. *A road matching method for precise vehicle localization using belief theory and Kalman filtering*, Autonomous Robots 19, pp.173-191, 2005.

[18] M. E. El Najjar and PH. Bonnifait. *Road Selection Using Multicriteria Fusion for the Road-Matching Problem*, IEEE Transactions on Intelligent Transportation Systems, 8, pp.279-291, 2007.

[19] D. Fox. *Adapting the sample size in particle filters through KLD-sampling*. The International Journal of Robotics Research, 22(12):985-1004, 2003.

[20] S. Gattein and P. Vannoorenberghe. *A comparative analysis of two approaches using the road network for tracking ground targets*, Seventh International Conference on Information Fusion, pp.62-69, 2004.

[21] A. Gelb. *Applied Optimal Estimation*. MIT Press, 1974.

[22] A. Gning and Ph. Bonnifait. *Dynamic vehicle localization using constraints propagation techniques on intervals.* a comparison with Kalman filtering. In IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005.

[23] A. Gning and Ph. Bonnifait. *Constraints propagation techniques on intervals for a guaranteed localization using redundant data*. Automatica, 42(7):1167–1175, 2006.

[24] Y.S. Greenfeld. *Matching GPS observations to localizations on a digital map*. Proceeding of the 81th annual Meeting of the Transportation Board, Washington D.C, 2002.

[25] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P. Nord-lund. *Particle filters for positioning, navigation, and tracking*. IEEE Transactions on Signal Processing, 50:425-435, 2002.

[26] Hao Xu; Hongchao Liu; Chin-Woo Tan; Yuanlu Bao. *Development and Application of an Enhanced Kalman Filter and Global Positioning System Error-Correction Approach for Improved Map-Matching*. Journal of Intelligent Transportation Systems: Technology, Planning, and Operations, vol.14, Issue:1, pp. 27-36, 2010.

[27] B. Hummel. *Dynamic and Mobile GIS: Investigating Changes in Space and Time*, chapter Map Matching for Vehicle Guidance. CRC Press, 2006.

[28] M. Jabbour, Ph. Bonnifait and V. Cherfaoui. *Map-Matching Integrity Using Multihypothesis Road-Tracking*, Journal of Intelligent Transportation Systems, 12, 189-201, 2008.

[29] L. Jaulin, M. Kieffer, O. Didrit and E. Walter. *Applied Interval Analysis*. Springer-Verlag, 2001.

[30] L. Jaulin. *Nonlinear bounded-error state estimation of continuous-time systems*. Automatica, 38, 1079-1082, 2002.

[31] W. Kim, G. Jee, J. Lee, *Efficient use of digital road map in various positioning for ITS*, IEEE Symposium on Position Location and Navigation, San Deigo, CA, 2000

[32] G. Nassreddine, F. Abdallah and T. Denœux. *Map matching algorithm using belief function theory*. In Proceedings of the 11th Int. Conf. on Information Fusion (FUSION'08), pages 995-1002, Cologne, Germany, June 30-July 03, 2008.

[33] G. Nassreddine, F. Abdallah and T. Denœux. *State estimation using interval analysis and belief function theory: Application to dynamic vehicle localization*. IEEE Transactions on Systems, Man and Cybernetics B, Vol 40 issue 5, pages 1205-1218, 2010.

[34] V. Noronha and M.Goodchild. *Map accuracy and location expression in transportation reality and prospects*. Transportation research C 8, 53-69, 2000.

[35] J.S. Pyo, D. H. Shin and T. K. Sung. *Development of a map matching method using the multiple hypothesis technique*. IEEE Intelligent Transportation Systems Conference Proceedings, pages 23-27, 2001.

[36] M.A. Quddus, W. Y. Ochieng , L. Zhao and R. B. Noland. *A general map matching algorithm for transport telematics applications*. GPS Solutions, Volume 7, Issue 3, pages 157-167, September 2003.

[37] M.A. Quddus, W. Y. Ochieng and R. B. Noland. *Integrity of map matching algorithms*. Transportation Research C, Volume 14, pages 283-302, November 2006.

[38] M.A. Quddus, R. B. Noland and W. Y. Ochieng. *A high accuracy fuzzy logic-based map-matching algorithm for road transport*. Journal of Intelligent Transportation Systems: Technology, Planning, and Operations 10 (3), pp. 103-115, 2008.

[39] G. Shafer. *A mathematical theory of evidence*. Princeton University Press, Princeton, N.J., 1976.

[40] E. Seignez, M. Kieffer, A. Lambert, E. Walter, T. Maurin. *Experimental vehicle localization by bounded-error state estimation using interval analysis.* Intelligent Robots and Systems conference, IROS 2005, pp. 1084-1089, 2-6 Aug. 2005.

[41] Sivaraman, S.; Trivedi, M.M.. *A General Active-Learning Framework for On-Road Vehicle Recognition and Tracking*. IEEE Transactions on Intelligent Transportation Systems, vol.11 Issue:2, pp. 267-276, 2010.

[42] Skog, I.; Handel, P.. *In-Car Positioning and Navigation Technologies A Survey*. IEEE Transactions on Intelligent Transportation Systems, vol.10, Issue:1 pp. 4-21, 2009.

[43] C. Smaili, M.E. El Najjar and F. Charpillet, *A Road Matching Method for Precise Vehicle Localization Using Hybrid Bayesian Network*, Journal of Intelligent Transportation Systems, 12, 176-188, 2008.

[44] Ph. Smets. The combination of evidence in the Transferable Belief Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(5):447-458, 1990.

[45] Ph. Smets and R. Kennes. *The Transferable Belief Model*. Artificial Intelligence, vol.66, pp. 191-234, 1994.

[46] S. Syed and M.E. Cannon. *Fuzzy logic-based map-matching algorithm for vehicle navigation system in urban canyons*. In proceedings of the Institute of Navigation (ION) national technical meeting, California, USA ,26-28 January, 2004.

[47] G. Taylor, G. Blewitt, D. Steup, S. Corbett and A. Car. *Road reduction filtering for GPS-GIS navigation*. Proc. of 3 AGILE conference on Geographic Information Science, Helsinki, Finland, pp. 114-120, 2001.

[48] Toledo-Moreo, R.; Betaille, D.; Peyret, F.. *Lane-Level Integrity Provision for Navigation and Map Matching With GNSS, Dead Reckoning, and Enhanced Maps*. IEEE Transactions on Intelligent Transportation Systems.vol. 11, Issue:1 pp. 100-112, 2010.

[49] G. Taylor, C. Brunsdon, J. Li, A. Olden,D. Steup and M. Winter. *GPS accuracy estimation using map-matching techniques: Applied to vehicle positioning and odometer calibration*. Computers, Environments, and Urban Systems, Vol. 30, pp. 757-772, 2008.

[50] C.E. White, D. Bernstein and A.L. Kornhauser. *Some map matching algorithms for personal navigation assistants*. Transportation Research Part C8,91-108, 2000.

[51] R. R. Yager and L. Liu, editors. *Classic Works of the Dempster-Shafer Theory of Belief Functions*. Springer, Heidelberg, 2008.

[52] L. Zhao, W.Y. Ochieng, M.A. Quddus and R.B. Noland. *An Extended Kalman Filter algorithm for Integrating GPS and low-cost Dead reckoning system data for vehicle performance and emissions monitoring*. Journal of Navigation, Vol. 56, pp. 257-275, 2003.

PLACE
PHOTO
HERE

Fahed Abdallah received the Dipl.Ing. and M.S. degrees in electrical and computer engineering from the Lebanese University, Beirut, Lebanon, in 1999 and 2000, respectively, and the Ph.D. degree in electrical and computer engineering from the Université de Technologie de Troyes, Troyes, France, in 2004. Since 2005, he has been an Associate Professor with the HEUDIASYC Laboratory, Université de Technologie de Compiègne, Compiègne Cedex, France. His current research interests involve state estimation for dynamic models based on multisensor fusion, statistical estimation and decision theories, pattern recognition, and belief-function theory.

PLACE
PHOTO
HERE

Ghalia Nassreddine was born in Makenh, Lebanon, in 1984. She received the Diploma in engineering from the Lebanese University, Tripoli, Lebanon, in 2005 and the M.S. degree and Ph.D. degree in computer science from the Université de Technologie de Compiègne, Compiègne Cedex, France, in 2006 and 2009, respectively. Her current research interests involve state estimation for dynamic models based on multisensor fusion, and belief function theory.

PLACE
PHOTO
HERE

Thierry Denœux graduated in 1985 as an engineer from the Ecole Nationale des Ponts et Chaussées in Paris, and received a doctorate from the same institution in 1989. Currently, he is Full Professor with the Department of Information Processing Engineering at the Université de Technologie de Compiègne, France. His research interests concern belief functions theory, fuzzy data analysis and, more generally, the management of imprecision and uncertainty in data analysis, pattern recognition and information fusion. He is the Editor-in-Chief of the *International Journal of Approximate Reasoning*, and a member of the editorial board of *Fuzzy Sets and Systems*.