# 2CoBel: A scalable belief function representation for 2D discernment frames 

Nicola Pellicanò, Sylvie Le Hégarat-Mascle, Emanuel Aldea

## To cite this version:

Nicola Pellicanò, Sylvie Le Hégarat-Mascle, Emanuel Aldea. 2CoBel: A scalable belief function representation for 2D discernment frames. International Journal of Approximate Reasoning, 2018, 103, pp.320-342. 10.1016/j.ijar.2018.10.007 . hal-01918407

## HAL Id: hal-01918407 <br> https://hal.science/hal-01918407

Submitted on 10 Nov 2018

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.

# 2CoBel: A Scalable Belief Function Representation for 2D Discernment Frames 

Nicola Pellicanò*, Sylvie Le Hégarat-Mascle, Emanuel Aldea<br>SATIE Laboratory, CNRS UMR 8029<br>Paris-Sud University, Paris-Saclay University, Orsay 91405, France


#### Abstract

This paper introduces an innovative approach for handling 2D compound hypotheses within the Belief Function framework. We propose a polygon-based generic representation which relies on polygon clipping operators, as well as on a topological ordering of the focal elements within a directed acyclic graph encoding their interconnections. This approach allows us to make the computational cost for the hypothesis representation independent of the cardinality of the discernment frame. For belief combination, canonical decomposition and decision making, we propose efficient algorithms which rely on hashes for fast lookup, and which benefit from the proposed graph representation. An implementation of the functionalities proposed in this paper is provided as an open source library. In addition to an illustrative synthetic example, quantitative experimental results on a pedestrian localization problem are reported. The experiments show that the solution is accurate and that it fully benefits from the scalability of the 2D search space granularity provided by our representation.

Keywords: Belief functions, 2D discernment frame, graph-based representation, localization


[^0]
## 1. Introduction

Belief Function Theory (BFT) [1, 2] is an increasingly popular framework for the generalization of probability and possibility theory by modeling imprecision and partial ignorance of information, in addition to its uncertainty.

BFT is widely used in fundamental tasks which benefit from multi-modal information fusion, such as object detection and data association for assisted driving [3, 4, 5], tracking [6, 7], object construction [8], outdoor localization (9), or autonomous robot mapping and tracking [10, 11], medical imaging [12], remote sensing [13], video surveillance [14], aircraft classification [15].

The main limitation, when dealing with such theory, since it copes with compound hypotheses, is the size of the set of hypotheses to handle, which may become intractable when the size of the discernment frame increases. Such issue becomes critical especially in higher dimensions, as when dealing with twodimensional (2D) spaces. In the rest of our paper, we define a set (in our case a frame or a focal set) as being a 2D set if its elements are elements of the Cartesian product of two totally ordered sets. Accordingly, a 2D discernment frame is defined as a frame of discernment which handles 2D focal elements. Now such 2D discernment frames can be encountered for instance in information fusion in the image domain (e.g., [8), in box particle filtering [7, or in localization

In [8], the authors aim to reconstruct objects from fragmentary detections in the image space. The discernment frame corresponds to the 2D image lattice. BFT is then exploited in order to perform object-detection data association, spatial extension of objects when new fragments are found, temporal conditioning for object displacement/disappear modeling and spatial conditioning for object separation modeling. Focal elements are represented as set of non-intersecting 2D-boxes.

In [9], 2D BFT is applied to the global navigation satellite system (GNSS) localization problem, where the information is represented by imprecise position measurements provided by several satellite sources, where complex focal
elements shapes (ring sectors) are modeled as sets of boxes.
In [11] the authors perform scene environmental mapping by making extensive use of evidential grids for spatial and temporal fusion, by converting the original 2D domain in a map of 1D BBAs.

While several public belief function theory libraries exist [16, [17, 18], all of them limited to 1 D representations, the use of 2 D spaces for information fusion has been recently explored for various tasks. In the following of this study, we focus on 2D discrete discernment frames. An exhaustive representation of $\Omega$ discrete hypotheses usually involves a discretization of the area as a grid, where each cell of the grid represents a singleton hypothesis [19, 11]. Focal elements are then expressed by using a binary word, where a bit equal to 1 means that the cell belongs to the focal set. Such straightforward binary-word representation of hypotheses allows for the definition of operators on sets through simple bitwise operations. However, such a representation suffers from major drawbacks when used in real world applications. Since there are $2^{|\Omega|}$ potential focal elements, large discernment frames become intractable, when the discretization resolution or the size of the whole area increases (different tasks may require different levels of precision for the solution, thus calling for a 2D space discretization which would increase quadratically the representation space size).

For such reasons, some works rely on different approaches to handle the 2D case: by proposing a smart sub-sampling of the 2D space to maintain tractability [19] by proposing a sparse representation of the set of hypotheses, and by keeping in memory only the ones which are carrying non-null information 9].

In order to make the representation manageable, [19] proposes to condition the detections acquired from one sensor in its field of view, and to perform a coarsening at a lower spatial resolution of the focal elements, depending on the physical properties of the sensor. While these workarounds help in practice, they do not make the application fully scalable with the size of the scene, and they involve approximations such as the already cited coarsening, or frequent
$6_{60} \mathrm{BBA}$ simplification, which aims at maintaining under control the number of focal elements of the BBAs.

Such limitations derive from the fact that the complexity of any basic operator between focal elements (e.g., intersection, union) depends on the cardinality of the focal elements themselves. The works in [8] overcome this limitation by different sets of boxes, which do not allow for fast focal element comparisons and lookup. Second, the box set representation implies a non-unique approximation of the real focal element shape once edges are not parallel to the axes of reference. Geometric approximations of such focal elements may require a very ${ }^{75}$ large set of boxes when precision is a concern. Moreover, subsequent operations involve increasing box fragmentation which may be detrimental both for performance and for memory load. In order to avoid deep fragmentation, in 9] some representation simplification procedures are presented, which in turn increase the cost of BBA management.

In [20], the authors rely on a description of multidimensional focal elements by discretizing their support as a point cloud, and by deriving an approximation of Dempster's rule by Monte Carlo simulation. They underline the fundamental issue raised by representations based on parametric functions which lead to difficult implementation of the elementary operations (intersection, union, and complementation), which are needed for Dempster-Shafer reasoning. Alternatively to this representation, in this paper we propose a novel approach, which overcomes the efficiency issues of parametric representations.

Following the idea of providing a sparse representation for 2D BFT, and motivated by the great benefit that an efficient representation would carry to
so high dimensional problems, we propose a new two-dimensional representation which has full scalability properties with respect to the size of the discernment frame, while allowing a theoretical infinite precision (bounded by the hardware
precision limitations).
Note that our purpose is different from theoretical studies in BFT [21] that

## 2. BBA representation

Let us denote by $\Omega$ the discernment frame, i.e., the set of mutually exclusive hypotheses representing the solutions. The power set $2^{\Omega}$ is the set of the $\Omega$ subsets having cardinality $2^{|\Omega|}$. The mass function $m$, specifying a BBA, is defined as $m: 2^{\Omega} \rightarrow[0,1]$ such that $\sum_{A \in 2^{\Omega}} m(A)=1$. A hypothesis $A \in 2^{\Omega}$ such that $m(A)>0$, is a focal element of $m$. The information encoded by a mass function can be represented in different equivalent formulations, most notably belief (Bel), plausibility ( $P l$ ) and commonality $(q)$ :

$$
\operatorname{Bel}(A)=\sum_{B \subseteq A} m(B), \quad P l(A)=\sum_{B \cap A \neq \varnothing} m(B), \quad q(A)=\sum_{B \supseteq A} m(B),
$$

defined $\forall A \subseteq \Omega$.
A BBA is said to be dogmatic if $\Omega$ is not a focal element, i.e., $m(\Omega)=0$.
A BBA is said to be consonant if the focal elements are nested: $\forall(A, B) \in$ $2^{\Omega} \times 2^{\Omega}, m(A)>0, m(B)>0 \Rightarrow A \subseteq B \vee B \subseteq A$.

In the following sections, two complementary representations are proposed.
In Section 2.1, a compact polygon-based geometric focal element description, for BBAs defined in 2D spaces, is detailed. It ensures low-level scalability for
primitive operators, while exhibiting fast comparison and lookup capabilities. It represents a generalization of the state-of-the-art representations [8, 19], which overcomes their limitations (outlined in Section 1).

In Section 2.2, a graph-based BBA description, independent from the BBA representation and from the space dimensionality, is proposed. It encodes the structural relationships between focal elements, and provides high-level scalability capabilities, e.g., for decomposition methods (Section 3.2) and for decision making algorithms (Section4). Graph construction, optimization and traversal strategies are discussed.

Let us consider a 2 D discernment frame $\Omega$. We will refer to the illustrative example in Figure 1 . Such an example, inspired by [19, represents a typical localization scenario, where the discernment frame is a bounded region representing the ground plane.

### 2.1. Focal element geometric representation

As mentioned in the Introduction, 2D focal element representations based on an exhaustive representation of $\Omega$ and binary words become intractable once the size of one axis of the discernment frame becomes greater than a few tens of units, and the box set representation [8] suffers from geometric approximation due to the fact that the number of boxes needs to be limited to small values.

In this work, we propose to represent the focal elements as generic polygons (or sets of polygons for focal elements having multiple components, e.g., focal elements with holes or which are split after a difference operation), by exploiting the capabilities of the generic 2D polygon clipping algorithms, efficient methods for basic operator implementations (intersection, union, difference and XOR). A focal element is represented by a set of closed paths, each of them represented by an ordered array of vertexes (counter-clockwise for positive areas, clockwise for holes). As operator implementations, we exploit an extension of Vatti's algorithm for clipping [23] implemented in the Clipper library [24].

The polygons are constrained to be simple, i.e., defined by closed simple paths (no crossing), with a minimum number of vertexes (no vertex joining two

(a)

(b)

$$
X^{*}=I_{1} \cap I_{2} \cap I_{4} \cap I_{5} \cap I_{7}
$$

(c)

Figure 1: Illustrative localization example. (a) BBA definition through its focal elements: camera detection $m_{1}$ (red), track at $t-1 m_{2}$ (green), road presence prior $m_{3}$ (blue), building presence mask $m_{4}$ (gray). (b) Focal elements obtained as a result of performing a conjunctive combination over the defined BBAs. (c) Intersection-inclusion graph and the result of graph simplification. The solid lines show the inclusion relationship, while the dashed lines highlight the intersection relationship. $X^{*}$ is the set with maximum $\operatorname{Bet} P$ value, retrieved as the result of the proposed Bet $P$ maximization method.


Figure 2: Example of representation of the focal element $P$ (containing a hole), as a set of polygons. The external and the internal circular paths are stored in counter-clockwise and clockwise directions respectively.
co-linear edges). Figure 2 shows an example of focal element representation as two polygons (one of them representing a hole). In case of multiple polygons per focal element, an additional constraint needs to hold: Given the set of paths composing a focal element, no edge of one path can cross an edge of another path. Under these constraints, the complexity of the basic operators between two polygons having $n$ and $m$ number of vertexes respectively, is $O(n m)$. Such lightweight representation presents also the advantages of uniqueness and precision. The set of (circular) vectors of vertexes of a focal element (set of polygons) provides a unique representation. The vertex coordinates use integer values for numerical robustness and correctness. This means that the continuous representation provided by polygons implies an underlining discretization. However, differently from the previous approaches, the coordinates can be rescaled at the desired level of precision (up to $\approx 10^{19}$ ) without any impact on the speed and memory requirements of the algorithm, being bounded only by the numerical representation limits of the hardware. This implies full scalability of the focal elements with respect to their size.

Example 1. Figure 1 1a shows an example of focal elements in the case of a localization application. The camera detection (red) is represented as a disk focal element, whereas the focal elements which have the shape of ring sectors embed the imprecision of the location and the ill-knowledge of the camera extrinsic pa-
rameters; the track (green) represents the location of the target at the previous
frame, whereas its dilation is used in order to model the imprecision in its position introduced by time; the gray and blue focal elements belong to two different $B B A$ s representing scene priors, of building and road presence respectively. The disk shaped focal elements are modeled as 64 to 128 vertexes regular polygons.

### 2.2. Graph-based representation

In the previous section, we have highlighted that an efficient geometric representation, as the one proposed, may lead to the definition of basic operators (e.g., intersection, union) which are cardinality independent (and thus scalable). However, such representation alone cannot guarantee full scalability properties when dealing, for example, with decision making algorithms, which work at singleton hypothesis level. Thus, claiming that a representation is spatially scalable requires some additional mechanisms which enforce scalability at an higher operational level than primitive operators on sets.

Indeed, many operations on BBAs, used for BBAs combination and decision making, can be expressed as algorithms which depend only on how the focal elements intersect with each other, irregardless of the actual shape or size of such elements. Typical use cases which need an efficient data structure, serving as the foundation for the proposal of efficient algorithms, are the canonical decomposition (Section 3.2) and the BetP maximization (Section 4).

Together with an efficient geometric representation, we propose a generic representation, independent from the actual geometric representation chosen, which expresses the relationships between the focal elements of a given BBA. In the following sections, we will propose first a straightforward representation based on graphs, then we will highlight its limitation, and, finally, we will propose some optimization techniques to improve its efficiency.

### 2.2.1. Straightforward graph definition and construction

We propose to encode the relevant topological links between the focal elements as an intersection-inclusion graph, i.e an intersection graph where an
edge is augmented in the case of a (directional) inclusion relationship.
Let us define the focal elements set as $\mathcal{A}=\left\{A_{1}, A_{2}, \ldots, A_{n}\right\}$. For optimization reasons explained further, the focal elements are labeled according to decreasing cardinality and the ordering follows the element label:

$$
\forall\left(A_{i}, A_{j}\right) \in \mathcal{A} \times \mathcal{A}, \quad i<j \Longrightarrow\left|A_{i}\right| \geq\left|A_{j}\right| .
$$

In the case of different focal elements with the same cardinality, the topological order is not unique. Thus, the graph representation itself, differently from the geometric one, is not unique in general. However, non-uniqueness, given the low-level polygon representation, is not a necessary property. Moreover, graph optimizations and graph-based algorithms (detailed further) do not require uniqueness as a prerequisite.

We build a directed acyclic graph (DAG) $G=(V, E)$ where each node $v \in V$ is a focal element and each edge $e \in E$ represents a non empty intersection between two focal elements, with the direction of the edge respecting the topological ordering. The inclusion relationship information is encoded into separate arrays. Each node has a reference to its including nodes with the lowest and highest label. For example, if focal element $A_{5}$ is included into $A_{1}, A_{3}, A_{4}$, then the node $v_{5}$ carries two pieces of information, namely $l_{5}^{l}=1$ and $l_{5}^{h}=4$, where $l_{j}^{l}$ and $l_{j}^{h}$ stands for lowest and highest label including focal element $A_{j}$.

Let us define the $k^{t h}$ path of length $m$ in the intersection-inclusion graph $G=(V, E)$ as $P_{k}^{(m)}=\left\langle v_{k, 1}, v_{k, 2}, \ldots, v_{k, m}\right\rangle$. Such path represents the intersection between all the focal elements related to the nodes included in the path. In the following, we will refer to the intersection derived from a path as the one computed from all its nodes.

Proposition 2.1. For any non empty intersection $I$ derived from a set $\widehat{A}$ of focal elements, there exists a path $P$ in the intersection-inclusion graph $G$, connecting the elements of $\widehat{A}$.

Proof. Let us consider $\widehat{A}=\left\{A_{1}, \ldots, A_{m}\right\}$ as the target set of focal elements
$\left(I=\bigcap_{A_{i} \epsilon \widehat{A_{i}}} A\right)$. It follows that, given the graph $G=(V, E)$ :

$$
\forall\left(A_{i}, A_{j}\right) \in \widehat{A} \times \widehat{A}, i<j \quad\left|A_{i} \cap A_{j}\right| \neq 0 \Rightarrow v_{i}, v_{j} \in V,\left(v_{i}, v j\right) \in E
$$

Since $I \neq \varnothing$, any node at index $i$ is connected to every node at index $j$, such that $j>i$.

The above formula implies:

$$
\forall v_{i} \in V,\left(v_{i}, v_{i+1}\right) \in E
$$

a sufficient condition for the existence of the path $P^{(m)}=\left\langle v_{1}, v_{2}, \ldots, v_{m}\right\rangle$.
Definition 2.1. A path $P_{k}^{(m)}$ is called dead if the intersection among the $m$
focal elements corresponding to its nodes is the empty set.
Definition 2.2. Given two paths $P_{k}^{(m)}$ and $P_{h}^{(n)}, P_{k}^{(m)}$ is called superpath of $P_{h}^{(n)}$ if $m>n$ and:

$$
\forall v \in P_{h}^{(n)} \Rightarrow v \in P_{k}^{(m)}
$$

Conversely, $P_{h}^{(n)}$ is called subpath of $P_{k}^{(m)}$.
Definition 2.3. A not dead path $P_{k}^{(m)}$, leading to the intersection $I_{k}$ is called redundant if there exists another path $P_{h}^{(n)}$ leading to the intersection $I_{h}$, such that $I_{h}=I_{k}$, and $P_{h}^{(n)}$ is a superpath of $P_{k}^{(m)}$.

225 In the following, we will refer to a path which is not redundant, equivalently, as a non-redundant path. While the graph structure can be used to explore all the possible intersections between focal elements, it shall be as efficient as possible in order to avoid exploring dead paths, while traversing only non-redundant paths, since they are by construction the paths carrying the greatest amount of structural knowledge (they gather all the focal elements which include the target intersection set).

### 2.2.2. Graph traversal and limitations

The determination of the useful paths (i.e., neither dead nor redundant) is performed through graph traversal. According to the chosen ordering (decreas235 ing cardinality), each node is iteratively selected as the root. For each root,
a depth first search strategy is used to traverse the graph. Then, given the current node $v_{i}$, the intersection between all the nodes of the current path is propagated as $I_{i}$; given an edge $e=(i, j)$, the node $A_{j}$ (for notation shortness a node is equivalently called by its represented focal element) is explored if ${ }_{240}\left|I_{j}\right|=\left|I_{i} \cap A_{j}\right|>0$. Such an operation is equivalent to performing a dynamic graph pruning which is a function of the current path, as soon as some branch leads to a dead path. Then, even if the number of node visits can be very large according to a brute force exploration, the dynamic pruning helps to cut out early dead paths, making the number of operations much lower in practice. In this form, however, the worst case for pruning happens with consonant BBAs, where every edge represents an inclusion relationship. In such case, the graph is complete, and the entire graph would need to be explored without any pruning possibility, even if most of the paths would be redundant. Such an observation, together with the fact that in practice consonant BBAs are widely used as ba250 sic representation of the initial imprecision increase with certainty, e.g., in the Dubois and Prade BBA allocation [25], justifies the use of the inclusion information for simplifying the graph and optimizing the traversal. In the following section, some optimization mechanisms will be proposed in order to reduce the number of explored paths during the traversal, thus decreasing the average com255 plexity of the algorithms which will be based on the graph representation. Three main sources of optimization will be presented.

### 2.2.3. Graph optimization

Proposition 2.2 (Root suppression). Given the current root node $v_{j} \in V$, if:

$$
\exists v_{i} \in V,\left(v_{i}, v_{j}\right) \in E, i<j \quad \text { s.t. } \quad A_{j} \mp A_{i},
$$

every path originating from root $v_{j}$ is redundant.
Proof. Let us consider a generic path starting at the root $v_{j}: P_{k}^{(m+1)}=\left\{v_{j}, v_{k, 1}, \ldots, v_{k, m}\right\}$. By definition of a DAG, the index of the nodes $v_{k, h}, h \in\{1 \ldots m\}$, is higher than $j$, and, thus, than $i$. Let us consider the corresponding set of focal elements

$$
\begin{aligned}
& \widehat{A}_{k}=\left\{A_{j}, A_{k, 1}, \ldots, A_{k, m}\right\}, \\
& I_{k}=A_{j} \cap\left(\bigcap_{h=1 \ldots m} A_{k, h}\right),
\end{aligned}
$$

and the augmented set $\widehat{A}_{k}^{\prime}=\left\{A_{i}, A_{j}, A_{k, 1}, \ldots, A_{k, m}\right\}$,

$$
I_{k}^{\prime}=\left(A_{i} \cap A_{j}\right) \cap\left(\bigcap_{h=1 \ldots m} A_{k, h}\right)=A_{j} \cap\left(\bigcap_{h=1 \ldots m} A_{k, h}\right)=I_{k} .
$$

Moreover, the path $P_{k}^{\prime(m+2)}=\left\{v_{i}, v_{j}, v_{k, 1}, \ldots, v_{k, m}\right\}$ is a superpath of $P_{k}^{(m+1)}$. It follows that $P_{k}^{(m+1)}$ is a redundant path.

Root suppression implies that only root nodes which correspond to focal elements not included in some others preceding them in topological order, can produce paths which are non-redundant. Thus, all the other nodes are suppressed as possible roots for the traversal. Such property justifies the choice of a topological sorting in descending order of cardinality. Algorithm 1 shows where root suppression is used. After constructing the graph, each node is taken into account as candidate root for depth first search. Root suppression is used in order to filter root candidates for the following graph traversal operations.

Proposition 2.3 (Early stopping). Given the current root $v_{r}$, a path containing a node $v_{j}$ included in a node $v_{h}, h<r$ is redundant.

Proof. Let us consider the generic path starting at the root $v_{r}$ and containing $v_{j}: P_{a, b}^{(m+n+2)}=\left\{v_{r}, v_{a, 1}, \ldots, v_{a, m}, v_{j}, v_{b, 1}, \ldots v_{b, n}\right\}$. The corresponding set of focal elements is $\widehat{A}_{a, b}=\left\{A_{r}, A_{a, 1}, \ldots, A_{a, m}, A_{j}, A_{b, 1}, \ldots A_{b, n}\right\}$ leading to the intersection $I_{a, b}$. Since $A_{j} \subset A_{h}$, we can define the augmented set $\widehat{A}_{a, b}^{\prime}=\left\{A_{h}, A_{r}, A_{a 1}, \ldots, A_{a, m}, A_{j}, A_{b 1}, \ldots A_{b, n}\right\}$, leading to the intersection:

$$
I_{a, b}^{\prime}=\left(A_{h} \cap A_{j}\right) \cap A_{r} \cap\left(\bigcap_{i=1 \ldots m} A_{a, i}\right) \cap\left(\bigcap_{i=1 \ldots n} A_{b, i}\right)=I_{a, b},
$$

since the term $\left(A_{h} \cap A_{j}\right)$ reduces to $A_{j}$.
Proposition 2.1 guarantees that the superpath $P_{a, b}^{\prime(m+n+3)}=\left\{v_{h}, v_{r}, v_{a, 1}, \ldots, v_{a, m}, v_{j}, v_{b, 1}, \ldots v_{b, n}\right\}$ exists. Thus, the path $P_{a, b}^{(m+n+2)}$

The early stopping criterion allows us to stop exploring a node if it is included in an already explored root. The constraint is equivalently expressed as the fact that early stopping is performed at $v_{j}$ if $l_{j}^{l}<r$ (among the nodes/focal elements that included $v_{j}$, indexed in $\left[l_{j}^{l}, l_{j}^{h}\right]$, there is at least one that has already been used as a root).

Early stopping is applied during graph exploration (see Algorithm 2 and Algorithm 3). It serves as a precondition for exploring or not a child of the current node.

Proposition 2.4 (Graph simplification). Given a node $v_{j}$ which has multiple incoming inclusion edges from $\left\{v_{i}^{h}\right\}_{h=1 \ldots m}$, all the edges but the one from the highest indexed node in topological order, $\left(v_{i}^{m}, v_{j}\right)$, belong to redundant paths.

Proof. First, one can demonstrate that, after removing the inclusion edges from $\left\{v_{i}^{h}\right\}_{h=1 \ldots m-1}, v_{j}$ is still reachable from $v_{i}^{1}$. Since the edge between $v_{i}^{m}$ and $v_{j}$ is kept, it is equivalent to demonstrate that $v_{i}^{m}$ is reachable from $v_{i}^{1}$.

$$
\begin{equation*}
\forall A_{i}^{k}, A_{i}^{s},(k, s) \in\{1, \ldots, m\}^{2}, A_{i}^{k} \cap A_{j}=A_{j}, A_{i}^{s} \cap A_{j}=A_{j} \Rightarrow\left|A_{i}^{k} \cap A_{i}^{s}\right| \geq\left|A_{j}\right| \neq 0 \tag{1}
\end{equation*}
$$

Thus, there exists a path from any node $\left\{v_{i}^{h}\right\}_{h=1 \ldots m-1}$ to $v_{i}^{m}$, and, consequently, to $v_{j}$.

Finally, we demonstrate that, if one of the removed edges is included in a path, that path is redundant.

Let us consider the path $P_{a, b}^{(n+q+2)}=\left\{v_{a, 1}, \ldots, v_{a, n}, v_{i}^{h}, v_{j}, v_{b, 1}, \ldots v_{b, q}\right\}$, where $h<m$, leading to the intersection $I_{a, b}$. Let us consider the node $v_{i}^{k}, h<k \leq m$, which, by topological ordering, cannot be already included into $P_{a, b}^{(n+q+2)}$.

Let us consider the superpath $P_{a, b}^{\prime(n+q+3)}=\left\{v_{a, 1}, \ldots, v_{a, n}, v_{i}^{h}, v_{i}^{k}, v_{j}, v_{b, 1}, \ldots v_{b, q}\right\}$, leading to the intersection $I_{a, b}^{\prime}$. Two edges have been added: $\left(v_{i}^{h}, v_{i}^{k}\right)$, guaranteed to exist by Equation $\mathbb{1}$; $\left(v_{i}^{k}, v_{j}\right)$, that exists by definition of the problem.

$$
I_{a, b}^{\prime}=A_{i}^{h} \cap\left(A_{i}^{k} \cap A_{j}\right) \cap\left(\bigcap_{i=1 \ldots n} A_{a, i}\right) \cap\left(\bigcap_{i=1 \ldots q} A_{b, i}\right)=I_{a, b},
$$

since the term $\left(A_{i}^{k} \cap A_{j}\right)$ reduces to $A_{j}$.
Thus, $P_{a, b}^{(n+q+2)}$ is a redundant path.

Graph simplification boils down to keeping, for each $v_{j}$, only the incoming inclusion connection from the node with the highest index in topological order $l_{j}^{h}$. Algorithm 4 details the graph construction steps, and shows how graph simplification is exploited. Any intersection edge between two nodes is added

However, after graph simplification, only the edges going from element $A_{i}$ to $A_{i+1}$ are kept, resulting into $k-1$ effective edges. Moreover, due to root suppression, only the first node will be used as root, so $k$ nodes in total will be explored, leading to $k$ non-redundant paths, providing $k$ different intersections, equal to the $k$ original focal elements.

| Node | $l^{l}$ | $l^{h}$ | Use as root | Deleted edges (simplification) |
| :---: | :---: | :---: | :---: | :---: |
| 1 | null | null | yes | - |
| 2 | 1 | 1 | no | - |
| 3 | 1 | 1 | no | - |
| 4 | 1 | 1 | no | - |
| 5 | 1 | 4 | no | $\left(v_{1} \rightarrow v_{5}\right),\left(v_{2} \rightarrow v_{5}\right)$ |
| 6 | 1 | 3 | no | $\left(v_{1} \rightarrow v_{6}\right),\left(v_{2} \rightarrow v_{6}\right)$ |
| 7 | 1 | 5 | no | $\left(v_{1} \rightarrow v_{7}\right),\left(v_{2} \rightarrow v_{7}\right),\left(v_{4} \rightarrow v_{7}\right)$ |

Table 1: Graph optimization main steps for the illustrative example in Figure 1

Example 2. We refer to the example in Figure 1. The graph on the left of Figure 1 illustrates the result of unoptimized graph construction. Intersection relationships are shown as dashed arrows, while inclusion relationships are depicted as solid arrows. The optimization steps are shown in Table 1. First,
$v_{1}$ includes all other focal elements, thus only $v_{1}$ will serve as a root for graph

## 3. BBAs combination

### 3.1. Classical combination rules and hashing

Numerous combination rules exist in order to mix the information provided by two sources. When the sources $m_{1}$ and $m_{2}$ are cognitively independent, the conjunctive combination rule is the most popular among them:

$$
\forall A \in 2^{\Omega}, m_{1} \odot m_{2}(A)=\sum_{\substack{(B, C) \in \mathcal{A}_{1} \times \mathcal{A}_{2}, B \cap C=A}} m_{1}(B) m_{2}(C),
$$

where $\mathcal{A}_{i}$ is the set of focal elements of $m_{i}$. In computational terms, the rule involves the construction of a new BBA by performing intersection operations between all pairs of focal elements from the two BBAs. According to the sum in the previous equation, when creating a new focal element from an intersection, one has to check for its existence and to add up some elementary mass product value if it already exists. The necessity of accumulating elementary masses into already existent focal elements (maybe computed with different operators than intersection, e.g., union in the case of the disjunctive rule), and thus to do an existence check every time a new mass value is computed, is not specific of the conjunctive rule, but it is shared with several other rules (e.g., the disjunctive rule [26], cautious rule [26] and evidential q-relaxation [27).

The above considerations justify the need for a BBA representation which allows for a fast lookup of a focal element in an array. The uniqueness and compactness of the proposed representation allow for an efficient and low collision prone hashing. The sparse set of focal elements of a given BBA can be
stored in a hash table, where the circular vector of vertexes is used to compute the hash. For a given polygon, its hash will be unique provided that we fix a BBA. On the contrary, polygon hashing can make direct use of the vertex data, thus not requiring any additional preprocessing step.

The hashing capability of the proposed representation may provide benefits
not only in case of BBAs combination rules, but for any operator which implies mass accumulation. Let us consider the case of coarsening [29]. Given two discernment frames $\Omega_{1}$ and a finer $\Omega_{2}$, let us define a refining function $\rho$ from $2^{\Omega_{1}}$ to $2^{\Omega_{2}}$, such that $\left\{\rho(\{o\}), o \in \Omega_{1}\right\}$ is a partition of $\Omega_{2}$ and $\forall A \subseteq \Omega_{1}, \rho(A)=$ $\bigcup_{o \in A} \rho(o)$. Let us consider the coarsening function $\rho^{-1}$. According to [29],
the least committed solution for defining $\rho^{-1}$ is the following outer reduction $\bigcup_{o \in A} \rho(o)$. Let us consider the coarsening function $\rho^{-1}$. According to [29],
the least committed solution for defining $\rho^{-1}$ is the following outer reduction function:

$$
\forall B \subseteq \Omega_{2}, \rho^{-1}(B)=\left\{o \in \Omega_{1} \text { s.t. } \rho(o) \cap B \neq \varnothing\right\}
$$

The BBA $m^{\Omega_{2} \downarrow \Omega_{1}}$ defined on $\Omega_{1}$ from a given BBA $m^{\Omega_{2}}$, defined on $\Omega_{2}$, and the coarsening function $\rho^{-1}$, is given by:

$$
\forall A \in 2^{\Omega_{1}}, m^{\Omega_{2} \downarrow \Omega_{1}}(A)=\sum_{B \subseteq \Omega_{2}, \rho^{-1}(B)=A} m^{\Omega_{2}}(B) .
$$

It follows that the derivation of $m^{\Omega_{2} \downarrow \Omega_{1}}$ can largely benefit from hashing, since policy to decide the starting vertex (e.g., the top left). The array hashing function is equivalent to the one implemented in the Boost library's [28] hash_range method.

The binary-word representation, in comparison, uses the full word as a unique key. However, the key length (in number of bits) grows linearly with the cardinality of the discernment frame, requiring the use of big data structures in order to store it. On the contrary, the proposed hash exhibits collision resistance property despite a fixed length. The box set representation 9, being not unique, does not allow for direct hashing without the extraction of the minimal set of vertexes on the boundary. A cheap alternative could be to hash the bounding box of the focal element, but this could cause frequent collisions, since it is common to have spatially close focal elements related to the same a focal element mass may be the result of several fragmented masses. Similar
deductions can be derived for operations such as conditioning [2], or marginalization of discernment frames defined as Cartesian products.

Example 3. Figure 16 illustrates the result of the conjunctive combination of the sources introduced in Figure 1a. Seven focal elements are produced.

### 3.2. Canonical decomposition and cautious rule

Canonical decomposition of a belief function allows us to represent a complex non-dogmatic $B B A{ }^{1}$ as the result of a combination (conjunctive combination will be referred throughout the discussion) of elementary belief states, namely Simple Support Functions (SSF) if the decomposed BBA is separable or ${ }_{370}$ a mixture of SSF and Inverse Simple Support Functions (ISSF) otherwise. The canonical decomposition, besides being a convenient representation for some combinations, has its interest into allowing the introduction of new combination rules, as the Denoeux cautious conjunctive rule, that is the least committed rule among conjunctive ones [26].

The decomposition of a non-dogmatic BBA, defined by Smets 30, uses the concept of generalized Simple Support Function (GSSF), defined as:

$$
\begin{aligned}
& \mu: 2^{\Omega} \rightarrow \mathbb{R}, \mu(A)=1-w \\
& \mu(\Omega)=w \\
& \mu(B)=0 \quad \forall B \in 2^{\Omega} \backslash\{A, \Omega\}
\end{aligned}
$$

where $A \neq \Omega$ and the weight $w \in \mathbb{R}^{+}$. The original BBA $m$ can be then expressed as a combination of basic GSSFs: $m=@_{A \subset \Omega} A^{w(A)}$. The conjunctive weight function $w(\cdot)$ is associated to any hypothesis included in discernment frame $\Omega$ :

$$
\ln w(A)=-\sum_{B \supseteq A}(-1)^{|B|-|A|} \ln q(B), \quad \forall A \subset \Omega
$$

${ }_{375}$ According to GSSF definition, only weights $w(A) \neq 1$ are useful for representing the original BBA (i.e., not leading to a vacuous GSSF). While consonant BBAs represent a special case for weights computation, where an iterative algorithm

[^1]with linear complexity in terms of the number of focal elements exists, the generic case raises computational problems.

In the 1D case, the fastest weight computation approach exploits the Fast Möbius Transform (FMT) [31] for transforming the $2^{|\Omega|}$ array of masses (dense representation) to a $2^{|\Omega|}$ array of commonalities $q$, and finally to a $2^{|\Omega|}$ array of weights. While this procedure is convenient for small $\Omega$ cardinalities, it is computationally infeasible for large discernment frames, like in the 2D case.

In this work, we propose to compute efficiently the canonical decomposition by constructing an ad-hoc discernment frame specific to the considered BBA. The idea is that, even if the considered 2 D discernment frame $\Omega$ is vast, for the considered BBA, the number of focal elements is limited and generally they are less than a few tens. Thus, the considered BBA can be represented on only a very restricted subpart of $\Omega$ with an effective spatial resolution that depends on the actual focal elements and will generally be much coarser than $\Omega$ resolution. Even more, the 2D structure of the focal elements is useless provided that the interaction properties between focal elements are preserved. Thus, for a given BBA, we aim to compute the coarsest possible equivalent representation where each element can be viewed as a unitary piece of information. If one retrieves this alternative representation, then it may be transformed into a 1D equivalent (by processing each of its elements as a singleton hypothesis) where the discernment frame is small, and thus suited for the FMT computation. Thus, the canonical decomposition no longer depends on the shape and original cardinality of the focal elements, but on their ad-hoc representation.

From $m^{\Omega}$ to coarse representation.

Definition 3.1. A set of disjoint sets $\mathcal{D}$ is a partition of the disjunction $\mathcal{X}=\cup_{A \in \mathcal{A}} A$ of the set of focal elements $\mathcal{A}$, where each element $D_{i}$ (namely, a disjoint set), satisfies:

$$
\forall D_{i} \in \mathcal{D},\left\{\begin{aligned}
D_{i} & \subset \mathcal{X}, \\
\left|D_{i} \cap D_{j}\right| & =0, \forall D_{j} \in \mathcal{D}, i \neq j \\
\forall A \in \mathcal{A},\left|D_{i} \cap A\right| \neq 0 & \Longleftrightarrow D_{i} \subseteq A
\end{aligned}\right.
$$



Figure 3: Illustrative example of coarse discernment frame extraction for canonical decomposition. (a) Initial BBA definition (focal elements are labeled); (b) Optimized intersectioninclusion graph for the given BBA : since $I_{4}$ is included in both $I_{1}$ and $I_{2}$, the edge between $v_{1}$ and $v_{4}$ has been deleted by graph simplification; (c) Final set of disjoint sets extraction (disjoint sets are labeled). In order to stress that the $x$ and $y$ axes are generic (dependent on the application), they are not labeled.

In addition, defining the set $\hat{A}_{i}$ :

$$
\hat{A}_{i}=\{A\}_{\substack{A \in \mathcal{A} \\ D_{i} \subseteq A}},
$$

as the set of all the focal elements including $D_{i}$, the following condition holds (maximal coverage):

$$
\forall D_{i} \in \mathcal{D}, \nexists D_{j} \in \mathcal{D}, i \neq j \text {, s.t. } \hat{A}_{i}=\hat{A}_{j}
$$

The coarsest possible representation consists in a subdivision of the discernment frame into a set $\mathcal{D}$ of disjoint sets. Basically, a set of disjoint set is the minimum cardinality set of non intersecting sets which do not cross the boundaries of any focal element. The set has minimal cardinality because of the maximal coverage constraint on each disjoint set.

In order to extract the set $\mathcal{D}$, one may exploit the graph representation introduced in Section 2.2. In graph terms, maximal coverage constraints guarantee that exactly one disjoint set for each non-redundant path can be constructed (thus, minimizing the number of disjoint sets). The graph traversal is conducted in a depth first search manner. Let us consider the addition of a new $D_{i}$ element to $\mathcal{D}$, that is the result of the intersection of all the focal elements in the path. For each $A_{k}$ in the explored path, a reference to the $i^{t h}$ disjoint set is stored in an auxiliary set of labels $S_{k}$. $S_{k}$ represents the set of all disjoint sets included into the focal element $A_{k}$. Now, when a new candidate $\widetilde{D}_{l}$ is found, it is not guaranteed to be disjoint from elements already present in $\mathcal{D}$. In order to extract the related set $D_{l}$ one has to apply the difference operator between $\widetilde{D}_{l}$ and any element already included in $\mathcal{D}$. The information regarding which disjoints sets are possibly included comes from the $S_{k}$ sets. If $\widetilde{D}_{l}$ is the result of the intersection along the path of length $m P_{l}=\left\{A_{l, 1}, \ldots, A_{l, m}\right\}$, the indexes $h_{l}$ of the disjoint sets to subtract can be retrieved as:

$$
h_{l}=\bigcap_{k=1 \ldots m} S_{l, k}
$$

representing the labels of all the disjoint sets included into $\widetilde{D}_{l}$. The resulting $D_{l}$ will be obtained as:

$$
D_{l}=\widetilde{D}_{l} \backslash\left\{D_{h}\right\}_{h \in h_{l}}
$$

If the sets are implemented as bit strings, the disjoint sets retrieval is as fast as $m$ bitwise operations. After the difference operation, the resulting disjoint set could be empty, and so ignored. Algorithm 2 provides a detailed outline of the set of disjoint sets extraction procedure through depth first search traversal.

|  | $\widetilde{D}=\bigcap_{i} I_{i}$ |  |  |  | D |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | $I_{1}$ | $I_{2}$ | $I_{3}$ | $I_{4}$ |  |
| A | $\checkmark$ | $\checkmark$ | $\checkmark$ |  | $D_{B}=\widetilde{D}_{B}$ |
| B | $\checkmark$ | $\checkmark$ |  | $\checkmark$ | $D_{C}=\widetilde{D}_{C} \backslash\left\{D_{A}, D_{B}\right\}$ |
| C | $\checkmark$ | $\checkmark$ |  |  | $D_{D}=\widetilde{D}_{D} \backslash\left\{D_{A}\right\}$ |
| D | $\checkmark$ |  | $\checkmark$ |  |  |
| E | $\checkmark$ |  |  |  | $D_{E}=\widetilde{D}_{E} \backslash\left\{D_{A}, D_{B}, D_{C}, D_{D}\right\}$ |
| F |  | $\checkmark$ |  |  | $D_{F}=\widetilde{D}_{F} \backslash\left\{D_{A}, D_{B}, D_{C}\right\}$ |

Table 2: Coarse representation computation from graph representation.

Example 4. Figure 3 shows the disjoint set decomposition on a 2 D didactic example. Each disjoint set corresponds to a subset of one or multiple focal elements which does not span over a focal element boundary. Table 2 specifies the disjoint set computation procedure, as $\mathcal{D}$ elements are extracted in the order assigned by the graph traversal. Each row corresponds to a path which possibly leads to a disjoint set. The candidate disjoint set $\widetilde{D}_{A}$, for example, is the result of $I_{1} \cap I_{2} \cap I_{3}$. Candidates are then transformed into actual disjoint sets by the difference operator. Let us specify the case of $\widetilde{D}_{D}$. Since $\widetilde{D}_{D}$ is the intersection between $I_{1}$ and $I_{3}$, we focus on the sets $S_{1}=\{A, B, C\}$ and $S_{3}=\{A\}$. In this example, the content of $S_{1}$ means that, among all the already extracted $\mathcal{D}$ elements, $D_{A}, D_{B}, D_{C}$ are included into $I_{1}$. The vector of indexes to subtract is computed as $h_{l}=S_{1} \cap S_{3}=\{A\}$. Thus, among all the already extracted $\mathcal{D}$ elements, $\widetilde{D}_{D}$ includes the disjoint set $D_{A}$, which has to be subtracted. Finally $D_{D}=\widetilde{D}_{D} \backslash\left\{D_{A}\right\}$. The intersections $I_{2} \cap I_{3}$ and $I_{2} \cap I_{4}$ are not explored, since both
$I_{3}$ and $I_{4}$ are included in $I_{1}$, which is an already explored root (early stopping). Moreover, for the same reason, nodes corresponding to $I_{3}$ and $I_{4}$ are not used as root (root suppression).

From the coarse representation to 1 D BBA. As soon as the $\mathcal{D}$ set is fully constructed, the sets $S_{k}$ serve to the conversion between the original BBA and a compact 1D representation. The new 1D BBA has a new discernment frame $\Omega^{\prime}$ of cardinality $|\mathcal{D}|$. The indexes of the disjoint sets represent the singleton hypotheses the 1 D BBA in $\Omega^{\prime}$. Each focal element $A_{k}$ is converted to a compound 1D hypothesis by using the $S_{k}$, which stores the indexes of the disjoint sets in $\mathcal{D}$ which, when performing a union operation, form the exact original set. Thus, each $A_{k}$ gives rise to:

$$
m^{\Omega^{\prime}}\left(\bigcup_{s \in S_{k}}\{s\}\right)=m^{\Omega}\left(A_{k}\right)
$$

Given that the cardinality of the ad-hoc 1D discernment frame $\Omega^{\prime}$ is lower than a few tens of elements in practical cases, it is then suitable for the applicacorresponding canonical decomposition sets are retrieved, the canonical decomposition of the original BBA can be obtained by mapping the 1D decomposition sets, expressed as union of singleton 1D hypotheses, to 2D sets, expressed as union of elements of $\mathcal{D}$

Example 5. We refer to the decomposition depicted in Figure 3. The new discernment frame is $\Omega^{\prime}=\{A, B, C, D, E, F\}$. All the original focal elements translate to focal elements in $\Omega^{\prime}$. For example, $m^{\Omega^{\prime}}(\{A, D\})=m^{\Omega}\left(I_{3}\right)$.

Such canonical decomposition approach is now fully scalable with the cardinality of the discernment frame, and it is especially convenient when the number of focal elements is much smaller than $|\Omega|$. Since the conjunctive canonical decomposition is trivially propagated when applying the conjunctive rule to two canonically decomposed BBAs, the typical scenario of applying the decomposition (e.g., for tracking), is at the BBA construction stage and when BBA
approximation is needed. At BBA construction stage, the number of focal elexploited for even faster computation. At BBA approximation stage, the number of focal elements is intentionally reduced, while the BBA is not consonant in general, making it the ideal scenario for the exploitation of the presented approach.

The canonical decomposition extraction approach allows us to use an alternative combination rule, which is particularly useful when the source cognitive independence assumption (assumed by the conjunctive rule) is not valid. The cautious rule 26 between two sources $m_{1}$ and $m_{2}$, with $w_{1}$ and $w_{2}$ associated canonical decomposition weight functions, is defined as:

$$
m_{1} ® m_{2}=\bigcap_{A \subset \Omega} A^{w_{1}(A) \wedge w_{2}(A)}
$$

where $\wedge$ denotes the minimum operator. In algorithmic terms a new canonical decomposition is built by including all the elements of the two initial decompositions with weight value lower than one. As for the case of the conjunctive rule, hashing can still be used for fast lookup of equal elements in the two decompositions.

### 3.3. Evidential q-relaxation

Recent work [27] introduces a BBA combination method which is robust to unreliable sources. The evidential $q$-relaxation, inspired by its equivalent in interval analysis (IA), allows us to relax a given number of sources when combining several belief functions. Let us denote $H_{r}^{N}$ the hypothesis that only $r$ out of $N$ sources are relevant, i.e., $q=N-r$ have to be relaxed. Let us call $\mathbf{A}=\left\{A_{1}, \ldots, A_{N}\right\}$, as an $N$-tuple of hypotheses, for $A_{i} \subset \Omega, i \in[1, N]$. Out of the $N$ hypotheses forming an N -tuple, only $r$ must be kept. Such meta-knowledge can be mapped as [27]:

$$
\Gamma_{\mathbf{A}}\left(H_{r}^{N}\right)=\bigcup_{\mathbf{A}^{\prime} \subseteq\left\{A_{1}, \ldots, A_{N}\right\},\left|\mathbf{A}^{\prime}\right|=r}\left(\bigcap_{A \in \mathbf{A}^{\prime}} A\right)
$$

For any element $B \subseteq \Omega$, its mass will be the sum, over all $\Gamma_{\mathbf{A}}\left(H_{r}^{N}\right)$ which are equal to $B$, of the products of masses of the focal elements of $A$ :

$$
\forall B \subseteq \Omega, m\left[H_{r}^{N}\right](B)=\sum_{\mathbf{A} \subseteq \Omega^{N}, \Gamma_{\mathbf{A}}\left(H_{r}^{N}\right)=B}\left[\prod_{i=1}^{N} m_{i}^{\Omega}\left(A_{i}\right)\right]
$$

Such a rule corresponds to a generalization of classic combination rules, since the special case of $r=N$ (i.e., do not relax any source) corresponds to the conjunctive rule, while the case $r=1$ corresponds to the disjunctive rule.

In computational terms, when considering the computation of the $\Gamma_{\mathbf{A}}\left(H_{r}^{N}\right)$ terms that appear combinatorial in terms of $N$ and $r$, the algorithm has higher time complexity than classic combination approaches, depending on the value of $q$. However, even for small values of $q$, such a method can dramatically improve the fusion performance in presence of outlier sources. Moreover, the proposed representation boosts (once more) the efficiency of the method, in terms of the efficient basic operators (the method makes heavy use of intersection and union operators), as well as of the use of hashing for fast accumulation of elementary masses.

## 4. Decision making

Once the different sources have been combined, the decision is generally taken on singleton hypotheses $\omega$ by maximizing the pignistic probability, defined as:

$$
\forall \omega \in \Omega, \operatorname{Bet} P(\omega)=\frac{1}{1-m(\varnothing)} \sum_{B \supseteq \omega} \frac{m(B)}{|B|}
$$

Even if the search space size is now $|\Omega|$, the decision making process is dependent on the cardinality of the discernment frame, and thus not scalable, limiting the precision level which can be set for a specific context.

In order to overcome this limitation, we propose a maximization algorithm which is independent from the cardinality of the sets, and which is only related to the number of focal elements in the BBA.

Definition 4.1. Given a set of focal elements $\mathcal{A}=\left\{A_{1}, \cdots, A_{n}\right\}$ a maximal intersection $I_{m}$ is derived from the set of focal elements $\tilde{\mathcal{A}} \subseteq \mathcal{A}$, such that any different focal element added to $\tilde{\mathcal{A}}$ would lead to an empty intersection:

$$
\begin{aligned}
& I_{m}=\bigcap_{A_{k} \in \tilde{\mathcal{A}}} A_{k}, \tilde{\mathcal{A}} \subseteq \mathcal{A},\left|I_{m}\right|>0 \text { s.t. } \\
& \nexists A_{s} \in \mathcal{A} \backslash \tilde{\mathcal{A}},\left|A_{s} \cap I_{m}\right|>0 .
\end{aligned}
$$

The underlying idea is that, since $B e t P$ is an additive measure, its maximum value can be achieved only for elements of the discernment frame which present maximal intersections.

The set $X^{*}$ of hypotheses that maximize the $\operatorname{Bet} P$ is researched within the set of maximal intersections $\mathcal{I}$ :

$$
X^{*}=\underset{I_{m} \in \mathcal{I}}{\arg \max } \frac{\operatorname{Bet} P\left(I_{m}\right)}{\left|I_{m}\right|},
$$

where the $B$ et $P$ function for compound hypotheses derives from the generalized formula:

$$
\forall A \in 2^{\Omega}, \operatorname{Bet} P(A)=\frac{1}{1-m(\varnothing)} \sum_{B \in \mathcal{A}, B \cap A \neq \varnothing} \frac{|A \cap B|}{|B|} m(B)
$$

of any other non-redundant path.

Proof. Let us assume that $I_{m}$ is redundant. Thus, there exists a superpath leading to the same intersection. Then, $I_{m}$ cannot be maximal.

Let us assume that $P_{m}$ is a subpath of another non-redundant path $P_{n}$, mo $n>m$. Thus, since non-redundant paths cannot be dead paths, $P_{n}$ leads to a non empty intersection. Then, $I_{m}$ cannot be maximal.

The graph-related definition of maximal intersection implies that any intersection not being located at a leaf of the dynamic graph cannot be a maximal intersection. The graph is said dynamic in the sense that a leaf is not only a node with no outgoing edges, but it is any node for which, given the current path, no outgoing edge can be explored further without leading to a dead path. Then, each leaf $l$ and the resulting $I_{l}$ is a candidate for maximal intersection. However, it could be non-maximal, as its associated set $\tilde{\mathcal{A}}$ could be a subset of a maximal intersection which has already been found. So, when a maximal intersection $I_{m}$ is found, the list $p_{m}$ of focal sets involving it is stored (using a bit-set representation). Once the new candidate $I_{l}$ is produced, the $p_{l}$ list is tested for inclusion against the stored candidates (by an AND operation between the bit-sets). Algorithm 3 provides full details on the depth first search strategy for the computation of maximal intersections.

| Path | Maximal intersection |
| :---: | :---: |
| $\left\langle v_{1} \rightarrow v_{2} \rightarrow v_{3} \rightarrow v_{6}\right\rangle$ | yes |
| $\left\langle v_{1} \rightarrow v_{2} \rightarrow v_{4} \rightarrow v_{5} \rightarrow v_{7}\right\rangle$ | yes |
| $\left\langle v_{1} \rightarrow v_{3} \rightarrow v_{6}\right\rangle$ | no |
| $\left\langle v_{1} \rightarrow v_{4} \rightarrow v_{5} \rightarrow v_{7}\right\rangle$ | no |

Table 3: Maximal intersection search details for the illustrative example in Figure 1

Example 6. Table 3 shows the intersection-inclusion graph traversal for maximal intersection search on the intersection-inclusion graph of the illustrative example of Figure 1. $X^{*}$ is selected as the one of the two maximal intersections at maximum BetP. For this example, raw traversal intersection graph would perform 42 node visits, while with the graph optimizations, presented in Section 510 2.2. 12 are executed. On the other hand, a straightforward BetP maximization by singleton hypothesis exploration would process 1100 locations (included into at least one focal element) with a factor 10 subsampling of the discernment
frame.

## 5. Experiments

We present test results on a synthetic toy example, as well as on a real tracking application scenario, which make use of the proposed representation, as well as of our publicly available 2 CoBel library, embedding all the described methodologies, and exploited throughout the entire testing.

### 5.1. The 2CoBel library

2 CoBel is an open sourc\& $\mathbb{Z}^{2}$ evidential framework embedding essential functionalities for generic BBAs definition, combination and decision making. An Evidence object defines common operations for a BBA containing any generic type of FocalElement. The current supported methods are: mass to Belief Functions conversion (plausibility, belief, commonality), conjunctive, disjunctive, cautious (exploiting the proposed canonical decomposition) rules and $q$ relaxation, vacuous extension and marginalization, conditioning, discounting, (generalized) BetP computation, BetP maximization (with singleton hypothesis enumeration or maximal intersections). Different representations of FocalElement are supported, each of them defining specifically basic operators (intersection, union, equality, inclusion): unidimensional (hashable), representing the 1D focal element as a binary string; $2 D$ bitmap, providing a bitmap representation as in [19]; 2D box set, implementing the definition and focal elements simplification operations proposed in 9]; 2D polygon (hashable), implementing our proposed representation.

The library has full support for discernment frames which are cartesian products.

[^2]
### 5.2. Case study: line estimation

As toy example for illustrating the applicability of the proposed representation to 2D domains, we tackle a fundamental problem in pattern recognition, namely the line estimation from a set of 2 D points. This example allows us to compare as well the effectiveness of the different combination rules implemented by the framework. Given a set of planar points in the $x y$ space, the objective is to infer the parameters of the line that fits at best the data. The Hough transform [32] is a classical approach to this problem, and the evidential framework allows us to handle the intrinsic imprecise and uncertain sources of the problem in the Hough domain. By using the polar representation of lines:

$$
\rho=x \cos \theta+y \sin \theta
$$

we build an accumulation space in the $(\rho ; \theta)$ domain in order to infer the values of the $\rho \in(-\infty,+\infty)$ and $\theta \in[0, \pi)$ parameters. The discernment frame $\Omega$ is then defined as a rectangular polygon in the $(\rho ; \theta)$ space. Since the $\rho$ parameter is unbounded, theoretically also $\Omega$ is an open set. However, in order to respect closed world assumptions, we bound $\Omega$ to extremely large values of $\rho$. Due to the sparse nature of the representation, the size of the discernment frame has no impact on the algorithm performance, so extreme values of $\rho$ equivalent to the max and min integer values supported by the hardware could be chosen.

Each data point $P_{i}=\left(x_{i}, y_{i}\right)$ votes for a family of lines which pass through it in the $x y$ plane. Each voted line $l_{j}^{i}=\left(\rho_{j}, \theta_{j}\right)$ corresponds to a point in the accumulation space. The locus of all the points in the accumulation space corresponds to a sinusoid function. Such toy example extends the one presented in [9, which performs straight line estimation in the $(\alpha ; \beta)$ space (where $y=$ $\alpha x+\beta)$. However, such space does not parametrize any possible line, and, moreover, a small possible interval of values has been considered in order to have a discernment frame of small size. On the one hand, the $(\alpha ; \beta)$ approach allows one to represent the constraints as straight lines rather than sinusoids, allowing for an inexpensive focal element representation. On the other hand,
the proposed representation allows us to move to a more convenient space where complex shapes can be defined.

In classical Hough approaches, since there is an infinite number of lines passing through the same point, the space $(\rho ; \theta)$ is quantized in such a way as to provide an acceptable precision. In the proposed representation, the resolution of the problem is given by the number of vertexes used to represent the polygons. However, the scalability of the problem allows us to rescale more flexibly the accumulation space for high precision estimations. In this experimentation we scale the accumulation space at a resolution of $10^{-2}$ for both $\rho$ and $\theta$ (in degrees).

The BBA construction consists in widening the sinusoidal function derived from each point in the dataset with imprecision and uncertainty knowledge. We build a consonant BBA having two focal elements. The first focal element is a sinusoidal band with width equal to $\delta \rho_{1}$, centered around the real sinusoidal function drawn from the data. Such focal element codes the imprecision of the points location given by the line discretization in the $x y$ space (typical if the space is in the image domain). The second focal element is a sinusoidal band with width equal to $\delta \rho_{2}>\delta \rho_{1}$, again centered around the real sinusoid. Such focal element encodes the uncertainty of the point location given by noisy data distribution.

Figure 4 shows several consonant BBAs (one for each data point) represented by polygonal approximations of sinusoidal bands in the accumulation space. Given $N$ points, the $N$ BBAs are then combined following some combination rule into a single BBA . The solution ( $\rho^{*}, \theta^{*}$ ) is finally obtained by performing Bet $P$ maximization on the output BBA . Since the result of the maximization is in general a closed area, and not a single point, the barycenter of the output polygonal result is considered as the proposed solution.

We test this evidential approach on simulated data, were $T=100$ lines are randomly drawn in the $x y$ space, and $M=10$ points for each line are extracted at fixed $x$ locations, under Gaussian noise assumption for the $y$ coordinate ${ }_{585} y_{i} \sim \mathcal{N}\left(-\frac{\cos \theta}{\sin \theta} x_{i}+\frac{\rho}{\sin \theta}, \sigma\right)$. Moreover, in order to evaluate its robustness to unreliable sources, the system is tested for different numbers of outliers ( 0,1


Figure 4: Example of BBAs construction for line estimation in the accumulation space. Every consonant BBA (one for each color), represents the information conveyed from a data point.
or 2 ), where an outlier is uniformly selected from the existing points and its $y$ value is shifted by a constant value $y_{0}$. In the proposed experiments, the BBAs parameters are set as $\delta \rho_{1}=2$ and $\delta \rho_{2}=6$, and the noise parameters are set as $\sigma=0.5$ and $y_{0}=5$.

The following different combination rules are evaluated: conjunctive rule, cautious rule, q-relaxation (with $q=1$ and $q=2$ ). The results obtained are compared with baseline least squares (LS), which is by definition the optimal estimator in presence of Gaussian noise. Figure 5 shows some line estimation examples extracted directly from the simulated data used for the quantitative evaluation.

Figure 6 shows the line estimation error distribution of the different methods under varying conditions, in terms of $\Delta \rho=\left|\rho^{*}-\rho_{g t}\right|$ and $\Delta \theta=\left|\theta^{*}-\theta_{g t}\right|$, where $\left(\rho_{g t}, \theta_{g t}\right)$ are the ground truth parameters for one line of the simulation dataset.

The performance of the various combination rules when the data is free from outliers are comparable with the optimal performance, in terms of the median error, achieved by standard LS. In such context, conjunctive and cautious rules give the same results, which is a desired property when the sources are independent. In the case of q-relaxation, even though the number of unreliable sources


Figure 5: Example of line estimation results for different numbers of outliers: (a) no outliers; (b) one outlier (conjunctive and cautious rule lines are identical); (c) two outliers; (d) line estimation in presence of correlated source data: cautious rule outperforms the other methods.


Figure 6: Radius error $\Delta \rho$ (first row) and angular error $\Delta \theta$ (second row), in presence of 0,1 or 2 outliers (from left to right), for the experiments on simulated data for the line estimation toy example. In each figure, from left to right, the bars correspond to: least squares (LS), conjunctive rule, cautious rule, q-relaxation $(q=1)$, q-relaxation $(q=2)$.解 same distribution. Thus, the derived BBAs are not independent. Moreover, the number of points composing the cluster represents a non-negligible percentage 35 of the total number of points ( $40 \%$ in the example). In this case, LS clearly
fails because univariate Gaussian noise assumption is violated, q-relaxation fails because it factors out few outliers, but it is still attracted by the rest of the cluster. Conjunctive rule fails because, when aggregating all the votes in the accumulation space, the cluster masses accumulate giving a strong weight to the estimation of the line from which they are drawn. Since the conjunctive rule is sensitive to the cluster size, it behaves estimating a wrong solution which tries to average the two lines. Conversely, cautious rule estimates the correct line accurately, because it processes the cluster of dependent points as a whole, thus not being influenced by the number of points composing it.


Figure 7: Example of disjoint sets decomposition on complex BBAs (obtained by iterative cautious combination of source BBAs); (a) after one combination; (b) after 2 combinations.

Figure 7 shows the disjoint set segmentation for efficient canonical decomposition (estimation of the ad-hoc 1D discernment frame, see Section 3.2), for a generic BBA as the one obtained by iteratively combining sinusoidal polygons. Such illustration points out that the computation of these disjoint sets is non trivial in general, while producing a segmentation of the discernment frame which is extremely convenient for canonical decomposition. In the presented scenario, since the starting BBAs are consonant, their canonical decomposition can be trivially computed as a special case at initialization time, and propagated after each cautious rule application as by definition.However, our aim here was to check the efficiency of our approach.


Figure 8: Example of pedestrian tracking steps. (a) Pedestrian detection blob. (b) Focal elements of detection $\mathrm{BBA} m_{d_{0}}$ on the ground plane at $t=0$ (the size of the largest focal element is approximatively $1 \times 2$ square meters). (c) Focal elements of the conjunctive combination $\widetilde{m}_{t_{0}, 7}$ between the track and the associated detection at $t=7$ ( 16 focal elements). (d) Focal elements of the BBA simplification of $\widetilde{m}_{t_{0}, 7}$ with the Jousselme's distance criterion ( 5 focal elements). (d) Focal elements after dilation of the track BBA $m_{t_{0}, 8}$ by polygon offsetting.

### 5.3. Case study: pedestrian tracking

We apply the proposed representation to the problem of tracking pedestrians detected by imprecise sensors, on the ground plane. The belief function framework allows for direct modeling of the imprecision associated with the detections and the tracks and provides a measure for data association between detections and tracks.

We make use of the detector proposed in 33, which performs low level information fusion from multiple cameras in order to provide a dense pedestrian detection map, together with pedestrian height estimations, in a range between 1.4 m and 2 m . The output of the detector allows us to project and track the detections on the ground plane. We demonstrate the use of the $2 D$ polygon representation provided in the 2 CoBel library in order to perform joint multiple target tracking in the Sparse sequence presented in [33]. We perform tracking on the provided detections for 200 frames of the Sparse sequence, and we measure the localization error of the real tracks (13 pedestrians, 4 standing and 9 moving) with respect to the ground truth. The tracker has to reconstruct lost tracks for given mis-detections occurring for up to six consecutive frames on the same pedestrian.

### 5.3.1. Discernment frame definition

The area under analysis is the ground plane region where the field of views of the cameras overlap. The area of the analysis region is $330 \mathrm{~m}^{2}$. The algorithm is run at a resolution of $10^{-4} \mathrm{~m}$, so that the cardinality of the discernment frame is $|\Omega|=33 \times 10^{9}$. While the desired localization precision is $10^{-2} \mathrm{~m}$, the chosen resolution is higher for increasing the robustness to rounding errors.

### 5.3.2. BBA construction and assignment

Given a detection $d_{i}$ at time $t$ located in $\left(x_{i}, y_{i}\right)$, we build a consonant BBA with two focal elements. The first focal element is a disk centered at $\left(x_{i}, y_{i}\right)$ and with a radius of 20 cm , taking into account the person's head and shoulder occupancy on the ground plane; the second focal element is a ring sector
(approximated by a trapezoidal shape), which embeds the height uncertainty

### 5.3.3. Data association and combination

Given a set of tracks at time $\delta, \mathcal{T}=\left\{t_{1}, \ldots, t_{k}\right\}$ and a set of detections $\mathcal{D}=\left\{d_{1}, \ldots, d_{h}\right\}$, the data association aims to compute an optimal one-to-one association solution $A_{l}=\left\{\left(t_{i}, d_{j}\right), i \in\{1 \ldots k\}, j \in\{1 \ldots h\}\right\}$ with respect to some defined cost. One $\left(t_{i}, \varnothing\right)$ association means that the track is into an inactive state (so it keeps propagating until it associates with a new detection or dies), while one $\left(\varnothing, d_{j}\right)$ association means a new track has to be initialized with detection $d_{j}$. We make use of the criterion in 34 to define the association cost:

$$
C_{t_{i}, d_{j}}=-\log \left(1-m_{t_{i}} @ m_{d_{j}}(\varnothing)\right),
$$

which expresses the data association task as a conflict minimization problem, which can be solved by the use of the Hungarian algorithm [35, 36].

The data association task is followed by a conjunctive combination which produces for every $\left(t_{i}, d_{j}\right)$ the new track:

$$
\widetilde{m}_{t_{i}, \delta}=m_{t i, \delta} \bigcirc m_{d_{j}} @ m_{p},
$$

where $m_{p}$ corresponds to the prior. It performs a masking operation on the visible region of interest of the camera on the ground plane.

A BBA simplification step is essential in tracking applications for two different reasons. First, we want to avoid that the number of focal elements grows without control as the time progresses, because it would mean that the real-time performance of the algorithm would degrade in time, bounding the maximum


Figure 9: Pedestrian tracking. (a) Detection blobs on the image space ( $t=0$ ) estimated by the detector in 33. Colors refer the estimated height values from $1.4 m$ (red) to $2 m$ (green). (b) Focal elements of the detection BBAs on the ground plane $(t=0)$. (c) Focal elements of track and detection BBAs on the ground plane $(t=8)$. Associated tracks and detections share the same color. (d) Final estimated tracks on first 20 frames. Red crosses refer to target locations, while colored sets correspond to regions presenting maximum Bet $P$ value.
number of processed frames. Second, we want to avoid an excessive fragmentation of the belief. The BBA simplification aims at reducing the number of focal elements of a given BBA while respecting the least commitment principle. We adopt the method proposed in [19], which chooses iteratively two focal elements to aggregate (by performing an union operation) as the ones which minimize the Jousselme's distance 37 between the original BBA and the summarized BBA, i.e., the one obtained after the aggregation.

The proposed representation allows, conversely to the one in 19 (which simplifies the BBA after each conjunctive combination), to perform the simplification on a less frequent time step. In the proposed experiment a target BBA is simplified when it reaches 15 focal elements, by producing a 5 focal element BBA.

### 5.3.5. Bet $P$ maximization

At each time step, we run the $\operatorname{Bet} P$ maximization algorithm presented in Section 4 for each active track $\widetilde{m}_{t_{i}, \delta}$ in order to extract the most probable location of the target. The cardinality of the resulting polygon represents the irreducible ambiguity in the target location. The target position is then estimated as the barycenter of the polygon.

### 5.3.6. Modeling the imprecision of the tracks prediction

Given the track $\widetilde{m}_{t_{i}, \delta}$, which represents the result of the conjunctive combination, we need to model the prediction step imprecision. In order to model the track displacement from the current location, a random walk term is added to the track. Such term boils down to an isotropic dilation of the focal elements. In the proposed representation, this corresponds to applying a scalable polygon offsetting algorithm, having $O(n \log n)$ complexity, where $n$ is the number of vertexes. Polygon offsetting allows for a dilation which respects the inclusion relationship of the original focal elements. The result of such step is the predicted track $m_{t i, \delta+1}$ at time $t+1$.

Figure 8 depicts an example of the proposed tracking steps for a single
pedestrian, specifically the BBA geometric representation after construction,

Figure 10: Normalized histogram of the localization error of pedestrian tracking on the Sparse sequence.

### 5.3.7. Results

Figure 9 show some qualitative results of pedestrian tracking in the Sparse sequence, highlighting the tracks estimated after the first 20 frames. In order to evaluate quantitatively the tracking accuracy, the target predicted locations are compared against an available ground truth. Such ground truth consists into coordinates in the image space where the heads are located. Since the height of such individuals is not known a priori, each location in the image space projects to a segment in the ground plane, allowing for any possible height in the interval of study. One computes the localization error as the distance between the target estimated location, and the ground truth head location, under the assumption that the height of such head corresponds to the predicted one. Such metric corresponds to computing the distance between the ground truth segment and a height uncertainty segment drawn at the target location. Target locations for inactive track states are estimated by linear regression fit of the estimated target positions at previous states.

Figure 10 shows the results in terms of (normalized) histogram of localization error. The average localization error is $\epsilon=0.2 \mathrm{~m}$, which reaches the empiric

| Resolution | Average Localization error |
| :---: | :---: |
| $10^{-1} \mathrm{~m}$ | 30.197 cm |
| $10^{-2} \mathrm{~m}$ | 22.340 cm |
| $10^{-3} \mathrm{~m}$ | 20.078 cm |
| $10^{-4} \mathrm{~m}$ | 19.944 cm |
| $10^{-5} \mathrm{~m}$ | 19.931 cm |

Table 4: Average localization error on the Sparse sequence using different discretization resolutions. By using a representation able to deal with finer resolutions, one may achieve a significant performance gain.
limit set by the intrinsic uncertainty of head spatial occupation. On the other hand, the average localization error remains steady in time, meaning that the estimated tracks do not tend to drift away from the real ones. The standard deviation of the average localization error in time is $\sigma=2.3 \mathrm{~cm}$.

Table 4 shows the average localization error obtained by the tracking algorithm for different choices of the resolution at which the discernment frame is discretized. When a coarse resolution of 10 cm is considered, the performance drops consistently. At this resolution the size of the discernment frame is already large enough to be intractable using methods based on binary representations, as in [19]. Moreover, while for the theoretically desired resolution of 1 cm the average localization error consistently drops, the proposed representation allows us to scale at finer resolutions to account for rounding errors, thus providing an additional performance boost.

For more complex tracking scenarios, the next step is to integrate the proposed representation in a more sophisticated model such as [7], which is supposed to cope with specific issues such as disambiguations or long term occlusions, and where our approach would extend box representations.

## 6. Conclusion

This paper proposed a new representation for multi-modal information fusion in bi-dimensional spaces in the BFT domain. Such representation exhibits uniqueness, compactness, space and precision scalability, which make it suitable for many settings constrained to large hypothesis spaces, where there is the need to extend the Belief Function framework with efficient multidimensional operators. In our experiments with actual data, we show the effectiveness of this formulation on multi-target tracking scenarios, where tenths of tracks have to be estimated on a wide region of interest. The main contributions can be summarized as follows:

- The proposal of a new polygon-based compound hypothesis representation, able to benefit from fast polygon clipping and hashing algorithms for scalability.
- The introduction of an intersection-inclusion directed acyclic graph to model the interaction between focal elements.
- The outline of efficient algorithms for the fundamental operators, decision making and decomposition methods which fully exploit the potential of both the geometric and graph representation proposed.
- The release of our contribution as a public library for the community, in order to ease the reproducibility of such representation for active research.

For future work, we are interested in developing a tracking algorithm for dense crowds, by performing cautious fusion of multiple detection sources provided by a smart camera network. We intend to highlight the interest of efficient 2D BFT representations for scaling such algorithms for high density crowds. In such contexts, the number of targets to track jointly can become intractable for state-of-the-art tracking frameworks, and an efficient evidential framework should fully benefit from the richer information provided by multiple cameras. Besides multi-target tracking applications, we also aim to study the impact
on our BBA representation when dealing with continuous discernment frames, since, despite some notable works, e.g., 38, 39, 40, belief functions have not because of computational limitations.

## A. Graph-based representation construction details

In the following, the construction of the intersection-inclusion graph from a given BBA is detailed. Algorithm 1 outlines the main steps for graph construction (specialized in Algorithm 4) and graph traversal (by iteratively executing either Algorithm 2 or Algorithm 3 at each root node).

```
Algorithm 1: Graph construction and traversal
    Data: Set of focal elements \(\mathcal{A}\) ordered by decreasing cardinality;
    begin
        \(\left(V, E, \mathbf{l}^{\mathbf{l}}, \mathbf{1}^{\mathbf{h}}\right)=\operatorname{BuildGRaph}(\mathcal{A}) ;\)
        for each \(v_{i} \in V\) do
            if \(l_{j}^{l}\) is not null then
                continue; //root suppression
            end
            \(\operatorname{DFS}\left((V, E), \mathcal{A}, i, \mathbf{l}^{\mathbf{l}}, v_{i}, A_{i},\{ \},\{ \},\{ \}\right) / /\) either Algorithm 2 or 3
        end
    end
```

```
Algorithm 2: DFS (SET OF DISJOINT SETS EXTRACTION)
    Data: DAG \(G=(V, E)\); set of focal elements \(\mathcal{A}\); root label \(r\); lowest
        including set label array \(\mathbf{l}^{\mathbf{l}}\); current node \(v_{i}\); current intersection
        \(I_{i}\); current path \(p\); sets of indexes of the disjoint sets included in
        each focal element \(S\); current set of disjoint sets \(\mathcal{D}\).
    Result: Set of disjoint sets \(\mathcal{D}\).
    begin
        \(p \leftarrow p \cup\left\{v_{i}\right\} ;\)
        for each \(e_{i j}=\left(v_{i} \rightarrow v_{j}\right) \in E\) do
            if \(l_{j}^{l}<r\) then
            continue; //early stopping
            end
            \(I_{j}=I_{i} \cap A_{j} ;\)
            if \(\left|I_{j}\right|>0\) then
            \(\operatorname{DFS}\left(G, \mathcal{A}, r, \mathbf{l}^{\mathbf{1}}, v_{j}, I_{j}, p, S, \mathcal{D}\right) ;\)
            end
        end
        \(h_{p}=\{1, \ldots,|\mathcal{D}|\} ;\)
        for \(v \in p\) do
            \(h_{p}=h_{p} \cap S[v] ; / /\) common disjoint sets among elements of the path
        end
        \(D=I_{i}\)
        for \(h \in h_{l}\) do
            \(D=D \backslash \mathcal{D}[h] ; / /\) subtract from \(D\) the included disjoint sets
        end
        if \(|D|>0\) then
            \(\mathcal{D} \leftarrow \mathcal{D} \cup\{D\} ;\)
            for \(v \in p\) do
                \(S[v] \leftarrow S[v] \cup\{|\mathcal{D}|\} ; / /\) focal element at node \(v\) contains \(D\)
            end
        end
        \(p \leftarrow p \backslash\left\{v_{i}\right\} ;\)
    end
```

```
        intersection \(P\); current set of maximal intersections \(\mathcal{I}\).
    Result: Set of maximal intersections \(\mathcal{I}\).
    begin
        \(p \leftarrow p \cup\left\{v_{i}\right\} ;\)
        leaf=true;
        for each \(e_{i j}=\left(v_{i} \rightarrow v_{j}\right) \in E\) do
            if \(l_{j}^{l}<r\) then
            continue; //early stopping
            end
            \(I_{j}=I_{i} \cap A_{j} ;\)
            if \(\left|I_{j}\right|>0\) then
                leaf= false;
                \(\operatorname{DFS}\left(G, \mathcal{A}, r, \mathbf{1}^{\mathbf{l}}, v_{j}, I_{j}, p, P, \mathcal{I}\right) ;\)
            end
        end
        if is a leaf then
            maximal= true;
            for \(p_{I} \in P\) do
            if \(p \subset p_{I}\) then
                maximal= false;
                break;
            end
        end
            if is maximal then
                \(\mathcal{I} \leftarrow \mathcal{I} \cup\left\{I_{i}\right\} ;\)
                \(P \leftarrow P \cup\{p\} ;\)
            end
        end
        \(p \leftarrow p \backslash\left\{v_{i}\right\} ;\)
    end
```

Algorithm 3: DFS (SET of MAXIMAL INTERSECTIONS EXTRACTION)
Data: DAG $G=(V, E)$; set of focal elements $\mathcal{A}$; root label $r$; lowest
including set label array $\mathbf{l}^{\mathbf{l}}$; current node $v_{i}$; current intersection
$I_{i}$; current path $p$; set of paths leading to each maximal

```
Algorithm 4: BUILDGRAPH
    Data: Set of focal elements \(\mathcal{A}\) ordered by decreasing cardinality;
    Result: Simplified DAG \(G=(V, E)\); lowest including set label array \(\mathbf{l}^{\mathbf{l}}\);
            highest including set label array \(\mathbf{l}^{\mathbf{h}}\);
    begin
        \(V=\{ \} ;\)
        \(E=\{ \} ;\)
        for each \(A_{i} \in \mathcal{A}\) do
            \(V \leftarrow V \cup\left\{v_{i}\right\} ;\)
            for each \(A_{j} \in \mathcal{A}, j>i\) do
            if \(A_{j} \subset A_{i}\) then
                \(l_{j}^{h}=i ;\)
                if \(l_{j}^{l}\) is null then
                    \(l_{j}^{l}=i ;\)
                end
                //delay inclusion edge storage (graph simplification)
            end
            else if \(\left|A_{i} \cap A_{j}\right|>0\) then
                \(E \leftarrow E \cup\left\{\left(v_{i} \rightarrow v_{j}\right)\right\} ; / /\) storing intersection edge
            end
                end
        end
        for \(j\) in \(1 \ldots|\mathcal{A}|\) do
            if \(l_{j}^{h}\) is not null then
                \(E \leftarrow E \cup\left\{\left(v_{l_{j}^{h}} \rightarrow v_{j}\right)\right\} ; / /\) storing inclusion edge
            end
        end
    end
```


## References

[1] A. P. Dempster, A generalization of Bayesian inference, in: Classic works of the Dempster-Shafer theory of belief functions, Springer, 2008, pp. 73-104.
[2] G. Shafer, A mathematical theory of evidence, Vol. 42, Princeton university press, 1976.
[3] R. O. Chavez-Garcia, O. Aycard, Multiple sensor fusion and classification for moving object detection and tracking, IEEE Transactions on Intelligent Transportation Systems 17 (2) (2016) 525-534.
[4] T. Denoeux, N. El Zoghby, V. Cherfaoui, A. Jouglet, Optimal object association in the Dempster-Shafer framework, IEEE transactions on cybernetics 44 (12) (2014) 2521-2531.
[5] R. Labayrade, D. Gruyer, C. Royere, M. Perrollaz, D. Aubert, Obstacle detection based on fusion between stereovision and 2d laser scanner, Pro Literatur Verlag, 2007.
[6] S. Hachour, F. Delmotte, D. Mercier, E. Lefèvre, Object tracking and credal classification with kinematic data in a multi-target context, Information Fusion 20 (2014) 174-188.
[7] T. A. Tran, C. Jauberthie, F. Le Gall, L. Travé-Massuyès, Evidential box particle filter using belief function theory, International Journal of Approximate Reasoning 93 (2018) 40-58.
[8] W. Rekik, S. Le Hégarat-Mascle, R. Reynaud, A. Kallel, A. Ben Hamida, Dynamic object construction using belief function theory, Information Sciences 345 (2016) 129 - 142. doi:https://doi.org/10.1016/j.ins. 2016. 01.047
[9] S. Zair, S. Le Hégarat-Mascle, Evidential framework for robust localization using raw GNSS data, Engineering Applications of Artificial Intelligence

61 (2017) 126 - 135. doi:https://doi.org/10.1016/j.engappai. 2017. 02.003
[17] T. Reineking, Dempster-Shafer theory library, https://pypi.python. org/pypi/py_dempster_shafer/0.7, last accessed 2018-03-13 (2014).
[18] A. Martin, Matlab toolbox for belief functions, http://www.arnaud. martin.free.fr/Doc, last accessed 2018-03-13 (2014). offsetting lines and polygons., http://www.angusj.com/delphi/clipper. php, last accessed 2018-03-13 (2014).
[25] D. Dubois, H. Prade, Representation and combination of uncertainty with belief functions and possibility measures, Computational intelligence 4 (3) (1988) 244-264.
[26] T. Denœux, Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence, Artificial Intelligence 172 (2-3) (2008) 234-264.
[27] F. Pichon, S. Destercke, T. Burger, A consistency-specificity trade-off to select source behavior in information fusion, IEEE transactions on cybernetics 45 (4) (2015) 598-609.
[28] Boost, Boost C++ Libraries, http://www.boost.org/, last accessed 2018-03-13 (2015).
[29] T. Denœux, A. Ben Yaghlane, Approximating the combination of belief functions using the fast moebius transform in a coarsened frame, International Journal of Approximate Reasoning 31 (1-2) (2002) 77-101.
[30] P. Smets, The canonical decomposition of a weighted belief, in: IJCAI, Vol. 95, 1995, pp. 1896-1901.
[31] R. Kennes, P. Smets, Computational aspects of the Mobius transformation, in: Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence, Elsevier Science Inc., 1990, pp. 401-416.
[32] P. V. Hough, Method and means for recognizing complex patterns, uS Patent 3,069,654 (Dec. 18 1962).
[33] N. Pellicanò, E. Aldea, S. Le Hegarat-Mascle, Geometry-Based Multiple Camera Head Detection in Dense Crowds, in: 28th British Machine Vision Conference (BMVC) - 5th Activity Monitoring by Multiple Distributed Sensing Workshop, Londres, United Kingdom, 2017.
URL https://hal.archives-ouvertes.fr/hal-01691761
[34] B. Ristic, P. Smets, The TBM global distance measure for the association of uncertain combat id declarations, Information fusion 7 (3) (2006) 276-284.
[35] J. Munkres, Algorithms for the assignment and transportation problems, Journal of the society for industrial and applied mathematics 5 (1) (1957) 32-38.
[36] R. Jonker, A. Volgenant, A shortest augmenting path algorithm for dense and sparse linear assignment problems, Computing 38 (4) (1987) 325-340.
[37] A.-L. Jousselme, D. Grenier, É. Bossé, A new distance between two bodies of evidence, Information fusion 2 (2) (2001) 91-101.
[38] B. Ristic, P. Smets, Belief function theory on the continuous space with an application to model based classification, in: Modern Information Process- ing, Elsevier, 2006, pp. 11-24.
[39] E. Ramasso, M. Rombaut, N. Zerhouni, Joint prediction of continuous and discrete states in time-series based on belief functions, IEEE transactions on cybernetics 43 (1) (2013) 37-50.
[40] A. Fiche, J.-C. Cexus, A. Martin, A. Khenchaf, Features modeling with an $\alpha$-stable distribution: Application to pattern recognition based on continuous belief functions, Information Fusion 14 (4) (2013) 504-520.


[^0]:    * Corresponding author

    Email addresses: nicola.pellicano@u-psud.fr (Nicola Pellicanò), sylvie.le-hegarat@u-psud.fr (Sylvie Le Hégarat-Mascle), emanuel.aldea@u-psud.fr (Emanuel Aldea)

[^1]:    ${ }^{1}$ Dogmatic BBAs are transformed into non-dogmatic ones by an $\epsilon$ discounting.

[^2]:    ${ }^{2}$ Implementation available at:
    https://github.com/MOHICANS-project/2CoBel

