



**HAL**  
open science

# Inference and decision in credal occupancy grids: use case on trajectory planning

Marie-Hélène Masson, Sébastien Destercke, Véronique Cherfaoui

## ► To cite this version:

Marie-Hélène Masson, Sébastien Destercke, Véronique Cherfaoui. Inference and decision in credal occupancy grids: use case on trajectory planning. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 2021, 29 (4), pp.537-557. <10.1142/S0218488521500239>. <hal-03341136>

**HAL Id: hal-03341136**

**<https://hal.science/hal-03341136v1>**

Submitted on 10 Sep 2021

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.



HAL Authorization

1 International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems  
2 © World Scientific Publishing Company

3 **Inference and decision in credal occupancy grids: use case on trajectory**  
4 **planning**

5 Marie-Hélène Masson.  
6 *UMR CNRS 7253 Heudiasyc*  
7 *Université de Picardie Jules Verne, France*  
8 *mmasson@hds.utc.fr*

9 Sébastien Destercke  
10 *UMR CNRS 7253 Heudiasyc*  
11 *Sorbonne Université, Université de Technologie de Compiègne*  
12 *CS 60319 - 60203 Compiègne cedex, France*  
13 *sebastien.destercke@hds.utc.fr*

14 Véronique Cherfaoui  
15 *UMR CNRS 7253 Heudiasyc*  
16 *Sorbonne Université, Université de Technologie de Compiègne*  
17 *CS 60319 - 60203 Compiègne cedex, France*  
18 *veronique.cherfaoui@hds.utc.fr*

19 Received (received date)  
20 Revised (revised date)

21 Occupancy grids are common tools used in robotics to represent the robot environment,  
22 and that may be used to plan trajectories, select additional measurements to acquire, etc.  
23 However, deriving information about those occupancy grids from sensor measurements  
24 often induce a lot of uncertainty, especially for grid elements that correspond to occluded  
25 or far away area from the robot. This means that occupancy information may be quite  
26 uncertain and imprecise at some places, while being very accurate at others. Modelling  
27 finely this occupancy information is essential to decide the optimal action the robot  
28 should take, but a refined modelling of uncertainty often implies a higher computational  
29 cost, a prohibitive feature for real-time applications. In this paper, we introduce the  
30 notion of credal occupancy grids, using the very general theory of imprecise probabilities  
31 to model occupancy uncertainty. We also show how one can perform efficient, real-time  
32 inferences with such a model, and show a use-case applying the model to an autonomous  
33 vehicle trajectory planning problem.

34 *Keywords:* Occupancy grids and Trajectory planning and Imprecision and Imprecise  
35 probabilities

36 **1. Introduction**

37 Occupancy grids are commonly used tools to represent a mobile robot (cars, drones,  
38 etc.) environment, especially autonomous ones. Based on a local perception of the

39 vehicle, they are instrumental in solving issues such as obstacle avoidance and short-  
40 term trajectory planning, and are particularly suitable in dynamic environments,  
41 for instance environments discovered on the fly by the robots (such as in SLAM  
42 problems <sup>27</sup>) or environments in which moving obstacles evolves <sup>19</sup>. Such grids  
43 are composed of multiple cells, which in the most usual settings can be either  
44 free or occupied. **Those grids are most of the time obtained from sensor readings**  
45 **associated to processing steps that may be peculiar to each sensor, according to**  
46 **their nature (cameras, lasers, ...)**. Such occupancy information is therefore prone  
47 to be affected by many sources of uncertainties, such as imprecise sensor readings  
48 (camera occlusion, numerical precision, ...), noise due to environmental conditions  
49 (rain, ...) or faulty sensors. Different frameworks have been proposed to model and  
50 reason from such uncertainties, for instance by using probabilistic <sup>21,6</sup>, fuzzy <sup>28,24,23</sup>  
51 or evidential <sup>20,33,26,30</sup> occupancy grids.

52 In this paper, we explore how another uncertainty framework, namely impre-  
53 cise probabilities, can be used to model uncertainty in occupancy grids. To our  
54 knowledge, it is the very first attempt to do so. Imprecise probability consists in  
55 extending the classical probabilistic model by considering convex sets of probabil-  
56 ities or equivalent models such as lower expectation. This representation formally  
57 includes all the other aforementioned representations <sup>10</sup> (possibilities, probabilities,  
58 belief functions), and has the benefit to be fully consistent with classical Bayesian  
59 probabilistic modelling, both axiomatically (as it relaxes Bayesian axioms) and for-  
60 mally speaking. It can therefore be interpreted either as an uncertainty theory of  
61 its own, or as a robustification of the classical probabilistic models. We refer to <sup>2</sup>  
62 for an introduction to various aspects of the framework, and to Walley's book <sup>32</sup>  
63 for a full theoretical account. In the case of cells taking values in the binary space  
64  $\mathcal{X} = \{x, \neg x\}$  composed of the states occupied ( $x$  in our notation) and free ( $\neg x$   
65 in our notation), imprecise probabilistic models encode uncertainty by explicitly en-  
66 coding imprecision through the use of two boundary measures, therefore allowing  
67 to differentiate between what is commonly called aleatoric uncertainty (due to a  
68 possible ambiguity and/or random phenomena) and epistemic uncertainty (due to  
69 an absence of information).

70 The concept of imprecise probabilistic occupancy grids is introduced in Section 2,  
71 where we also discuss the associated challenges in terms of inference and decision.  
72 We then consider in Section 3 the specific problem of making inferences for local  
73 trajectory planning, proposing a computationally efficient and simple solution to do  
74 so. Each step of the method is illustrated using simple examples. Then, in Section 4,  
75 for illustrative purposes, we present some results obtained using real occupancy grids  
76 coming from PACPUS <sup>17</sup>, an experimental platform developed in our laboratory,  
77 dedicated to researches on autonomous vehicles.

## 78 2. Introducing credal occupancy grids

79 In this section, we will introduce the notion of credal occupancy grids, and will  
80 discuss the problem of making general inferences with them. Roughly speaking, such  
81 inferences come down to make computations over (pseudo-)Boolean functions where  
82 the probabilities of literals are imprecisely specified. We will however not treat this  
83 very general issue that is encountered in many fields such as data-bases<sup>13</sup>, game-  
84 theory<sup>25</sup>, but will focus on the most common issue in applications using occupancy  
85 grids, that is guiding a moving object through it.

86 For this reason, we will consider from now that our problem concerns a moving  
87 robot, and that the robot environment is modelled by a large grid of cells, and that  
88 each cell  $X$  can take a binary value, occupied ( $x$ ) or unoccupied ( $\neg x$ ). We assume  
89 that when a robot perceives its environment through measurement devices (sensors,  
90 cameras), it receives uncertain information about whether a given cell is occupied.  
91 Imprecise probabilities<sup>32</sup> offer a very generic way to model uncertainty, including  
92 most known models of uncertainty such as precise probabilities, belief functions or  
93 possibility distributions<sup>10</sup>. It also comes with a fully-fledged theory to reason with  
94 such uncertainty and make decisions<sup>32</sup>.

95 In a binary setting such as the occupancy of cell  $X$ , any imprecise probabilistic  
96 model can be reduced to an interval  $[p(x), \bar{p}(x)]$  of the probability to be occupied,  
97 as we have that  $p(\neg x) = 1 - \bar{p}(x)$  and similarly for the upper bound. **Note that the**  
98 **fact that any convex probability set can be reduced to such intervals on singletons**  
99 **is only true in the binary case, and is used here as one of the key feature that**  
100 **makes our approach tractable.** We can in particular model the following extreme  
101 situations:

- 102 • certain occupancy:  $[p(x), \bar{p}(x)] = [1, 1]$ ,
- 103 • certain inoccupancy:  $[p(x), \bar{p}(x)] = [0, 0]$ ,
- 104 • unknown occupancy:  $[p(x), \bar{p}(x)] = [0, 1]$ ,

105 as well as all the intermediate ones, with precise probabilities being retrieved when  
106  $p(x) = \bar{p}(x)$ . **A credal occupancy grid is pictured in Figure 1, where cells are volun-**  
107 **tary of very large size for illustrative purposes. Green cells are likely to be unoccu-**  
108  **pied, red ones occupied, and grey/dark ones are in between (i.e.,  $0.5 \in [p(x), \bar{p}(x)]$ ).**  
109 **The darker the shades, the wider are the corresponding intervals.**

Let us now consider a set  $X_1, \dots, X_m$  of cells. As they are equivalent to Boolean  
variables, any event  $A$  concerning these cells (e.g., “at least one cell is occupied”, “ $k$   
consecutive cells are occupied”) can be expressed as a Boolean formula, and in  
particular can be put in Disjunctive Normal Form (DNF). This means that we  
can express any event in the form of a disjunction  $A = \bigvee_{i=1}^j A_i$ , where each  $A_i =$   
 $\bigwedge_{k \in A_i^+} x_k \wedge \bigwedge_{k \in A_i^-} \neg x_k$  is a conjunction of values for the cells ( $A_i^+$  denotes the set of  
indices of positive variable,  $A_i^-$  the indices of negative one). Such sets are sometimes  
called orthopairs<sup>5</sup> in the literature. When the uncertainty over cells is expressed  
by a precise probability, assessing  $P(A)$  then amounts to evaluate  $P(\bigcup_{i=1}^j A_i)$ , a

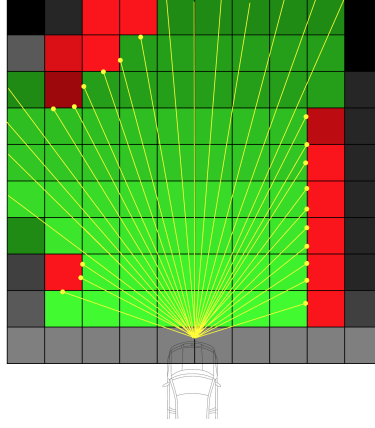


Fig. 1. Credal occupancy grid build from sensor data. Certain occupancy:  $[p(x), \bar{p}(x)] = [1, 1]$  in red, certain inoccupancy:  $[p(x), \bar{p}(x)] = [0, 0]$  in green, unknown occupancy:  $[p(x), \bar{p}(x)] = [0, 1]$  in black. The intermediate states are represented with lower intensity values.

potentially difficult task. One common way to do it is to use the following formula<sup>a</sup>

$$P(A) = P(\cup_{i=1}^j A_i) = \sum_{\mathcal{A} \subseteq \{A_1, \dots, A_j\}} -1^{|\mathcal{A}|} P(\cap_{A \in \mathcal{A}} A_i) \quad (1)$$

110 where the second equality simply applies the inclusion/exclusion principle. The  
 111 probability of a conjunctive event  $A_i$  is

$$P(A_i) = \prod_{k \in A_i^+} p(x_k) \prod_{k \in A_i^-} p(\neg x_k). \quad (2)$$

112 and likewise for intersection of such events, that remain of the same form. Solving  
 113 Equation (1), hence making inferences over a probabilistic grid, is then relatively  
 114 straightforward when probabilities are precise, as assessing the probabilities of con-  
 115 junctions is easy. It should however be noted that there are multiple ways to decom-  
 116 pose a given event  $A$  into a DNF, such as the use of Binary Decision Diagrams<sup>4</sup> or of  
 117 minimal cuts in reliability theory<sup>16</sup>, that can lead to more or less computationally  
 118 friendly decompositions, in the sense that the number of terms in the right-hand  
 119 side of (2) can grow exponentially. Obtaining compact Boolean representations of  
 120 an event  $A$  is therefore a whole area research, that is outside of the scope of the  
 121 present paper, which is why we will focus on the second part on specific cases for  
 122 which efficient decompositions do exist.

123 However, even when one has obtained a decomposition through some means,  
 124 the task of solving Equation (2) becomes much more difficult when probabilities

<sup>a</sup>Note that in this paper, we will use interchangeably  $\vee$  and  $\cup$ , or  $\wedge$  and  $\cap$ , as they have the same meaning.

125 become imprecise, as we then have to solve the equation

$$\underline{P}(A) = \inf_{p(x_i) \in \{\underline{p}(x_i), \bar{p}(x_i)\}} P(A) \quad (3)$$

126 to get the lower bound of the probability of an event  $A$ , which is NP-hard in general,  
127 as Equation 1 is a multi-linear form of the  $p(x_i)$ , each constrained to be in an interval  
128 (see for instance <sup>3</sup>). Indeed, if it is known <sup>12</sup> that the bound given by Equation 3  
129 is obtained in such a case on one of the vertices of the hyper-cube  $\times_i [\underline{p}(x_i), \bar{p}(x_i)]$ ,  
130 that is  $p(x_i) \in \{\underline{p}(x_i), \bar{p}(x_i)\}$ , it remains to find which one, hence to check for  $2^m$   
131 combinations if  $m$  is the number of cells. The upper bound can be found similarly,  
132 replacing inf by sup, with the same difficulties. This means, in particular, that  
133 reasoning over an entire imprecise probabilistic grid and for arbitrary events will be  
134 computationally prohibitive and infeasible for real-time applications.

**Example 1.** Let us consider two cells  $X_1, X_2$ , and the event “one and only one cell is occupied”, that corresponds to a XOR logical gate, that is

$$A = (x_1 \wedge \neg x_2) \vee (x_2 \wedge \neg x_1) = A_1 \cup A_2$$

with  $A_1 \cap A_2 = \emptyset$ , as each contains opposite values (e.g.,  $x_1$  for  $A_1$ , and  $\neg x_1$  for  $A_2$ ). Let us now consider that  $p(x_1) \in [0.6, 0.8]$  and  $p(x_2) \in [0.3, 0.6]$ , we have that  $\underline{P}(A)$  is obtained on a combination of extreme points, i.e.,

$$\begin{aligned} \underline{P}(A) &= \inf_{p(x_i) \in \{\underline{p}(x_i), \bar{p}(x_i)\}} p(x_1) (1 - p(x_2)) + (1 - p(x_1)) p(x_2) \\ &= 0.8 \cdot 0.4 + 0.2 \cdot 0.6 = 0.44 \end{aligned}$$

135 obtained for  $p(x_1) = 0.8$  and  $p(x_2) = 0.6$  (the upper bounds of  $p(x_1)$  and  $p(x_2)$ ).

136 More generally, one may be interested not only in computing the lower and  
137 upper bounds of some event probabilities, but to compute the lower and upper  
138 expected values of some reward/penalty function  $f : \times_{k=1}^m \{x_k, \neg x_k\} \rightarrow \mathbb{R}$ . Since it  
139 is defined on a discrete space, we can always define a partition  $B_1, \dots, B_k$  such that  
140  $f$  is constant over elements of this partition, i.e. if we denote by  $\mathbf{x}$  an assignment  
141 vector over cells  $X_i$ ,  $f(\mathbf{x}) = f(\mathbf{x}')$  whenever  $\mathbf{x}, \mathbf{x}' \in B_i$  for some  $i \in \{1, \dots, k\}$ . It  
142 then makes sense to denote by  $f_{B_i}$  the value of  $f$  for all elements included in  $B_i$ .  
143 Computing the lower expectation of  $f$  than amounts to solve the equation

$$\underline{\mathbb{E}}(f) = \inf_{p(x_i) \in \{\underline{p}(x_i), \bar{p}(x_i)\}} \sum_{j=1}^k f_{B_j} P(B_j) \quad (4)$$

144 with each  $B_j$  that can be decomposed using (2). As (4) remains a multi-linear form,  
145 the problem for solving it is in general not more difficult than the one of finding a  
146 lower probability, and its tractability will depend on both the form of  $f$  and of the  
147 events  $B_i$ .

148 **Example 2.** Let us pursue Example 1, by considering a simple game: we get a  
149 reward of 2 if the two variables end up with the same value, and suffer a penalty of

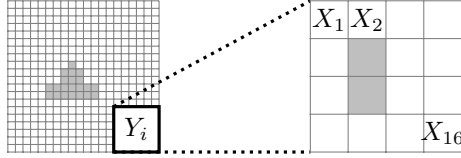


Fig. 2. The metagrid  $Y_i$  composed of 16 cells of the credal occupancy grid

150 1 (equivalently, a reward of  $-1$ ) if they have different values. Our knowledge about  
 151 the variables  $X_1, X_2$  remains the same. One may think, for instance, of a game with  
 152 two coins with ill-known probabilities of landing head and tail.

We now want to know our lower expected reward, should we play this game. We have  $B_1 = (x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2)$  and  $B_2$  its complement, with  $f_{B_1} = 2$  and  $f_{B_2} = -1$ . We must now minimize

$$2(p(x_1) \cdot p(x_2) + p(\neg x_1) \cdot p(\neg x_2)) - (p(\neg x_1) \cdot p(x_2) + p(x_1) \cdot p(\neg x_2))$$

153 which gives  $\underline{\mathbb{E}}(f) = -0.86$ , obtained for the values  $p(x_1) = 0.8$  and  $p(x_2) = 0.6$ .

154 In the rest of the paper, we will propose a decision model that focuses on partic-  
 155 ular types of events, that are of practical interest in robot guidance problems and  
 156 are computationally tractable, in the sense that bound (3) can be computed in a  
 157 time that is linear in the number of considered cells (or Boolean variables).

### 158 3. Moving in credal occupancy grid: model, inference and decision

159 Now that we have introduced credal occupancy grids and discussed their main  
 160 aspects regarding inference, let us focus on the problem of guiding a moving robot  
 161 within such an occupancy grid.

162 From now on, we will also be interested in particular states of subgrids of the  
 163 whole occupancy grid, that we will call metagrids. Such metagrids will describe the  
 164 successive locations of a robot along a chosen path. Indeed, usually the space occu-  
 165 pied by a robot (be it a vehicle, a drone, etc.) is much bigger than a single cell, whose  
 166 area is usually related to the smallest zone a robot can perceive with sufficient reli-  
 167 ability. As an example, cells of grids for autonomous vehicles are typically squares of a  
 168 few decimetres. A metagrid  $Y_i$  will be composed of  $n$  cells denoted  $X_1, \dots, X_n$ , while  
 169 a possible path for the robot will always be composed of  $k$  metagrids  $Y_1, \dots, Y_k$ ,  
 170 representing its successive positions. **The concept of metagrid within a grid is shown**  
 171 **in the Figure 2 (with a metagrid being made of 16 elementary cells).**

172 Given this, our decision model will be based on the following elements, that we  
 173 will detail in different subsections:

- 174 • For each metagrid  $Y_i$ , assess the uncertainty over the fact that at least  
 175 one cell of the metagrid is occupied: this provides lower/upper bounds  
 176  $[\underline{p}(y_i), \bar{p}(y_i)]$  over a new Boolean state (either at least one cell is occupied,

- 177 or none), that we consider as probability bounds of the fact that the robot  
178 can move over  $Y_i$ .
- 179 • Assess the lower/upper probability of the  $k$  mutually exclusive events  
180  $F_1, \dots, F_k$  that the robot can move up to the  $i$ th metagrid, that is  
181  $F_i = \bigwedge_{j=1}^{i-1} \neg y_j \wedge y_i$ , corresponding to the event that  $Y_i$  is the first occu-  
182 pied metagrid.
  - 183 • Pick the trajectory that provides the best expected reward as the robot  
184 future path, unless stopping is required because the first metagrids are too  
185 likely to be occupied.

186 For each of those steps, we will provide inference tools whose computational com-  
187 plexity is fixed and linear with the number of states, hence ensuring that real-time  
188 applications of the proposed method is doable. Since they also rely on fixed formu-  
189 las, they can also be easily implemented as hardware components, possibly speeding  
190 up computations even further.

### 191 3.1. Inference about individual metagrids

A first inference we want to do is to know, for each metagrid, what is the likelihood of having at least one cell  $X_i$  that is occupied, as having one such cell is judged sufficient for the metagrid to be unreachable. In practice, this comes down to assess the probability bounds for the event

$$Y = x_1 \vee x_2 \vee \dots \vee x_n.$$

One advantage of this event is that it satisfies a monotonicity constraint, i.e., it cannot become false if one variable  $X_i$  goes from the value  $\neg x_i$  to  $x_i$ . **It results that the corresponding multi-linear form given by Equation (2) is increasing<sup>b</sup> in each variable  $p(x_i)$ . Previous results about such multi-linear forms<sup>12</sup>, applied for instance in<sup>1</sup> to the case of reliability functions, then indicate that  $\underline{P}(Y)$  is obtained by computing  $P(Y)$  when  $p(x_i) = \underline{p}(x_i)$ .** Yet it should be noticed that the form  $x_1 \vee x_2 \vee \dots \vee x_n$  is not very computationally friendly, as every pair of elements of the disjunction have a non-empty intersection, since no pair of formulas contains opposite values of variables (each formula is only made of positive values). This means that Equation (1) contains a number of terms that factorially increases with  $n$ . However, we can re-express  $Y$  as a disjunction of non-intersecting events (using, e.g., ordered binary decision diagrams<sup>4</sup>)

$$Y = x_1 \vee (\neg x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2 \wedge x_3) \vee \dots \vee (\bigwedge_{i=1}^{n-1} \neg x_i \bigwedge x_n)$$

192 that counts  $n$  different conjunctions that have the advantage to be pairwise disjoint,  
193 meaning that we finally get

<sup>b</sup>One can simply consider the partial derivatives  $\partial P(A)/\partial p_i$  to check it.

	$X_1$	$X_2$	$X_3$	$X_4$
$\underline{p}(x_i)$	0.2	0.1	0	0.6
$\bar{p}(x_i)$	0.2	1	0.1	0.7

Table 1. Example 2: a metagrid with 4 cells

$$\underline{P}(Y) = \sum_{i=1}^n \left( \underline{p}(x_i) \prod_{j=1}^{i-1} \bar{p}(\neg x_j) \right),$$

194 a sum that has a linear number of terms with respect to  $n$ . We can similarly obtain  
 195 an upper bound  $\bar{P}(Y)$ , just inverting the bounds over  $p(x_i)$  in the equation. The  
 196 bounds  $[\underline{P}(Y^c), \bar{P}(Y^c)]$  over the event  $Y^c$ —“none of the cells of the metagrid are  
 197 occupied” can then be obtained by complementation, as we have  $\underline{P}(Y^c) = 1 - \bar{P}(Y)$   
 198 and  $\bar{P}(Y^c) = 1 - \underline{P}(Y)$ .

**Example 3.** Let us consider a metagrid composed of four cells with probability intervals of occupancy given in Table 1. In this case, we have:

$$\begin{aligned} \underline{P}(Y) = & \underline{p}(x_1) + \underline{p}(x_2)\bar{p}(\neg x_1) \\ & + \underline{p}(x_3)\bar{p}(\neg x_1)\bar{p}(\neg x_2) \\ & + \underline{p}(x_4)\bar{p}(\neg x_1)\bar{p}(\neg x_2)\bar{p}(\neg x_3). \end{aligned}$$

Using the fact that  $\bar{p}(\neg x_i) = 1 - \underline{p}(x_i)$ , we obtain:

$$\begin{aligned} \underline{P}(Y) = & 0.2 + 0.1(1 - 0.2) \\ & + 0(1 - 0.2)(1 - 0.1) \\ & + 0.6(1 - 0.2)(1 - 0.1)(1 - 0) = 0.712 \end{aligned}$$

Similarly,

$$\begin{aligned} \bar{P}(Y) = & \bar{p}(x_1) + \bar{p}(x_2)\underline{p}(\neg x_1) \\ & + \bar{p}(x_3)\underline{p}(\neg x_1)\underline{p}(\neg x_2) \\ & + \bar{p}(x_4)\underline{p}(\neg x_1)\underline{p}(\neg x_2)\underline{p}(\neg x_3). \end{aligned}$$

Thus, we have:

$$\begin{aligned} \bar{P}(Y) = & 0.2 + 1(1 - 0.2) \\ & + 0.1(1 - 0.2)(1 - 1) \\ & + 0.7(1 - 0.2)(1 - 1)(1 - 0.1) = 1, \end{aligned}$$

199 giving a probability interval of occupancy equal to  $[0.712, 1]$ , what might indicate  
 200 that the metagrid is not free. Note that the upper bound is equal to 1 which is quite  
 201 natural because it is totally plausible for the second cell to be occupied. Similarly,  
 202 the high lower bound is induced by the fact that the fourth cell is probably occupied.

	$Y_1$	$Y_2$	$Y_3$	$Y_4$
$\underline{P}(y_i)$	0.1	0.3	0.712	0
$\overline{P}(y_i)$	0.2	0.5	1	1

Table 2. Example 3: a trajectory composed of four metagrids

### 203 3.2. Inferences along a trajectory

Having made inferences over the  $k$  metagrids  $Y_1, \dots, Y_k$  along a given trajectory, we are now concerned about selecting the path that the robot should take. A first step for this is to assess how far we could go along a given trajectory. Indeed, if one of the first metagrid is almost surely occupied, it is not very useful to know that the last ones are surely free, as we will have encountered an obstacle before them. Hence, what matters is how soon we are likely to encounter an obstacle on a path. With this reasoning in mind, we propose to assess the lower and upper probability bounds over consecutive events  $F_i$ ="the  $i$ th metagrid is the first non-free", or in variable terms,

$$F_i = \bigwedge_{j=1}^{i-1} \neg y_j \wedge y_i \quad i = 1, k.$$

The event  $F_i$  being actually expressible as a Cartesian product of the subspaces  $\{y_i, \neg y_i\}$ , we have in the probabilistic case that

$$P(F_i) = p(y_i) \prod_{j=1}^{i-1} p(\neg y_j).$$

204 Getting bounds over this probability is then very easy, since if all the terms are  
205 independent, it is enough to just minimize each of them. We thus end up with

$$\underline{P}(F_i) = \underline{p}(y_i) \prod_{j=1}^{i-1} \underline{p}(\neg y_j) \quad (5)$$

206 and similarly for  $\overline{P}(F_i)$ , just switching the bounds. Computing this bound involves  
207 at most  $k$  products, and is therefore quite efficient again. Note that we compute  
208 also the bounds for an additional event  $F_{k+1} = \bigwedge_{j=1}^k \neg y_j$  which represents the event  
209 "all metagrids are free".

210 **Example 4.** Let us consider a trajectory composed of 4 metagrids which prob-  
211 abilities are given in Table 2 (in which the third metagrid could be the one of  
212 Example 3).

These values could correspond to the following situation: the first two metagrids are likely to be free, then the sensor detects almost surely an obstacle on the third metagrid, and does not see anything on the last metagrid because of occlusions. We have:

$$F_1 = y_1 \times \{y_2, \neg y_2\} \times \{y_3, \neg y_3\} \times \{y_4, \neg y_4\}$$

	$F_1$	$F_2$	$F_3$	$F_4$	$F_5$
$\underline{P}(F_i)$	0.1	0.24	0.28	0	0
$\overline{P}(F_i)$	0.2	0.45	0.63	0.18	0.18

Table 3. Example 3: bounds computation for the  $F_i$ 

$$F_2 = \neg y_1 \times y_2 \times \{y_3, \neg y_3\} \times \{y_4, \neg y_4\}$$

$$F_3 = \neg y_1 \times \neg y_2 \times y_3 \times \{y_4, \neg y_4\}$$

$$F_4 = \neg y_1 \times \neg y_2 \times \neg y_3 \times y_4$$

$$F_5 = \neg y_1 \times \neg y_2 \times \neg y_3 \times \neg y_4$$

213 Using Esq. (5), we can compute easily the probability bounds of the five events.  
214 They are given in Table 3.

It should also be noted that the events  $F_1, \dots, F_{k+1}$  actually form a partition of the space  $\prod_{i=1}^k \{y_i, \neg y_i\}$  ( $F_i \cap F_j = \emptyset$  for any  $i, j$ , and  $\cup_{i=1}^{k+1} F_i = \prod_{i=1}^k \{y_i, \neg y_i\}$ ), meaning that they can be interpreted as probability bounds over atoms of a Boolean algebra. This means in particular that the set of intervals

$$[\underline{P}(F_i), \overline{P}(F_i)], \quad i = 1, k$$

215 can be interpreted as probability intervals<sup>9</sup>, a well known imprecise probabilistic  
216 model for which efficient inference tools do exist. We will use this fact in the next  
217 section, when deciding which path a mobile robot should follow in order to maximize  
218 its utility.

### 219 3.3. Deciding the best trajectory

220 We now consider that we have to choose between several paths  $\tau$ , each moving  
221 over a fixed number of  $k$  metagrids whose states are uncertain. To choose the best  
222 trajectory, we propose to use a utility function quantifying the fact that a trajectory  
223 is usable or not. This utility is defined on the basis of the events  $F_i$ .

224 A first remark is that the states or atoms  $F_i$ ,  $i = 1, k + 1$  of our decision space  
225 are naturally ordered in terms of increasing utility, in the sense that  $F_j$  (the  $j$ -th  
226 metagrid is the first occupied) happening is worse than  $F_{j+1}$  happening. This means,  
227 among other things, that the utility function associated to those states should be  
228 increasing, i.e., that  $u(F_i) \leq u(F_{i+1})$  in any case.

In the case of precise probabilities (we have a precise value  $P_j(F_i)$  for each element of a trajectory  $\tau_j$ ), the value of a trajectory would then be its expected utility  $\mathbb{E}(\tau_j)$ , i.e.

$$\mathbb{E}(\tau_j) = \sum_{i=1}^{k+1} P_j(F_i)u(F_i).$$

229 When the  $P(F_j)$ 's become imprecise and are only known to belong to some inter-  
 230 vals  $[\underline{P}(F_j), \overline{P}(F_j)]$ , this expectation becomes interval-valued for each trajectory  $\tau_j$ .  
 231 Given the fact that our uncertainty model is expressed as intervals on singletons,  
 232 the upper and lower probabilities are 2-monotone and thus we can compute the  
 233 lower and upper expected utilities of each trajectory thanks to a Choquet integral  
 234 (we refer to <sup>9</sup> for the details of the derivation). Moreover, as the utilities are always  
 235 ordered in the same way, computing the bounds of the expected utility is simple as  
 236 we have:

$$\mathbb{E}(\tau_j) = \sum_{i=1}^{k+1} (u(F_i) - u(F_{i-1})) \underline{P}(\{F_i, \dots, F_{k+1}\}),$$

237 and

$$\underline{P}(\{F_i, \dots, F_{k+1}\}) = \max\left(\sum_{l=i}^{k+1} \underline{p}(F_l), 1 - \sum_{l=1}^{i-1} \overline{p}(F_l)\right),$$

238 with  $u(F_0) = 0$ . Similarly

$$\overline{\mathbb{E}}(\tau_j) = \sum_{i=1}^{k+1} (u(F_i) - u(F_{i-1})) \overline{P}(\{F_i, \dots, F_{k+1}\}),$$

239 and

$$\overline{P}(\{F_i, \dots, F_{k+1}\}) = \min\left(\sum_{l=i}^{k+1} \overline{p}(F_l), 1 - \sum_{l=1}^{i-1} \underline{p}(F_l)\right).$$

240 Again, due to our assumptions (increasing and ordered utilities along the  $F_i$ , that  
 241 form a partition), these formulas are fixed and straightforward to compute.

**Example 5.** Let us consider the utilities given in Table 4. We have chosen to  
 fix negative utilities for the two first metagrids, representing a kind of “security  
 distance” of at least two metagrids, a negative expected utility then corresponding  
 to the fact that if this path is chosen, the mobile robot should preferably brake or  
 stop. If we compute the lower and upper bounds for the trajectory of Example 4,  
 we have

$$\begin{aligned} \underline{\mathbb{E}}(\tau) &= (-20 - 0) \cdot \underline{P}(\{F_1, \dots, F_5\}) + (-10 - (-20)) \cdot \underline{P}(\{F_2, \dots, F_5\}) + \dots \\ &\quad + (20 - 10) \cdot \underline{P}(\{F_5\}) \\ &= (-20 - 0) \cdot 1 + (-10 - (-20)) \cdot 0.8 + \dots + (20 - 10) \cdot 0 = -8.5 \end{aligned}$$

242 and we finally find  $[-8.5, 1]$ , which corresponds to the fact that the second metagrid  
 243 is likely to be occupied.

244 As each trajectory is characterized by a lower and an upper expectation, there  
 245 are several possibilities to make a decision about the best trajectory, as we must now

$u(F_0)$	$u(F_1)$	$u(F_2)$	$u(F_3)$	$u(F_4)$	$u(F_5)$
0	-20	-10	0	10	20

Table 4. Example 3 continuation: utility values

246 compare intervals and no longer point-valued evaluations. Some classical ways <sup>31</sup> to  
247 do so are the following:

- 248 • Order 1. We can say that a trajectory  $\tau_j$  is better than a trajectory  $\tau_k$   
249 ( $\tau_j \succ \tau_k$ ) if and only if  $\underline{\mathbb{E}}(\tau_j) \geq \bar{\mathbb{E}}(\tau_k)$ .
- 250 • Order 2. We can say that a trajectory  $\tau_j$  is better than a trajectory  $\tau_k$  if  
251 and only if  $\underline{\mathbb{E}}(\tau_j) \geq \underline{\mathbb{E}}(\tau_k)$  and  $\bar{\mathbb{E}}(\tau_j) \geq \bar{\mathbb{E}}(\tau_k)$ .
- 252 • Order 3. We can say that a trajectory  $\tau_j$  is better than a trajectory  $\tau_k$  if  
253 and only if  $\underline{\mathbb{E}}(\tau_j) \geq \underline{\mathbb{E}}(\tau_k)$ .
- 254 • Order 4. We can say that a trajectory  $\tau_j$  is better than a trajectory  $\tau_k$  if  
255 and only if  $\bar{\mathbb{E}}(\tau_j) \geq \bar{\mathbb{E}}(\tau_k)$ .

The two first rules provide a partial order on the trajectories allowing for incomparabilities, whereas the two last rules provide a linear order. Rule 3 is a pessimistic rule, it consists in retaining the best trajectory in the worst case, i.e. the trajectory associated to the highest lower expected utility. Rule 4 on the contrary gives the preference to the highest upper expected utility, adopting in this way an optimistic behaviour. When the order is partial (rules 1 and 2), one can determine a set of non-dominated trajectories:

$$T^* = \{\tau_j \mid \nexists \tau_k \text{ such that } \tau_k \succ \tau_j\}$$

256 If this set is composed of more than one trajectory and that a unique trajectory  
257 is mandatory, several strategies can then be applied. It is for example possible  
258 to postpone the decision by asking complementary information on the navigable  
259 space. It is also possible to use additional criteria in the decision process, like the  
260 proximity to a reference trajectory (using in this case what is commonly known as  
261 lexicographic orders <sup>11</sup>).

262 **Example 6.** Let us suppose that we have 5 possible trajectories, whose lower and  
263 upper expected utilities are given in Table 5 and represented in Figure 3. Using Rule  
264 1, there are three maximal elements in the partial order induced by the intervals  
265 ( $\tau_2$ ,  $\tau_4$ , and  $\tau_5$ ). Two maximal elements are found using Rule 2 ( $\tau_4$  and  $\tau_5$ ). The  
266 pessimistic rule 3 leads to the choice of  $\tau_4$ , whereas Rule 4 chooses  $\tau_5$ .

#### 267 4. Illustrative application

268 As a proof of concept, we experiment our approach using real data, issued from the  
269 platform PACPUS <sup>17</sup> developed in our lab. The objective of PACPUS is to provide

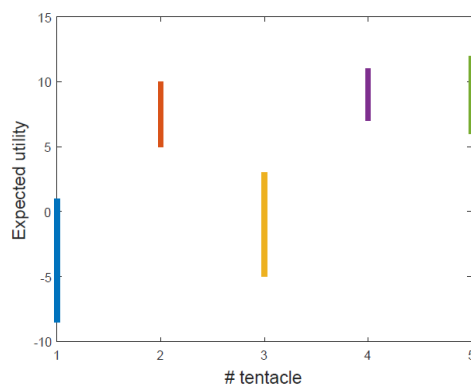


Fig. 3. Representation of the expected utilities of the trajectories

	$\tau_1$	$\tau_2$	$\tau_3$	$\tau_4$	$\tau_5$
$\underline{\mathbb{E}}(\tau_j)$	-8.5	5	-5	7	6
$\overline{\mathbb{E}}(\tau_j)$	1	10	3	11	12

Table 5. Bounds of expected utilities of the trajectories

270 tools and resources for experimenting on intelligent vehicles. We use a Lidar as a  
 271 perception sensor. This sensor can distinguish between free and occupied space and  
 272 model it in 2D (x,y coordinates) with respect to the vehicle bodyframe. We used the  
 273 **credal** grids generated by a C++ code<sup>19</sup> with data acquired on our experimental  
 274 vehicle (cf Figure 4). **We refer to the provided references for further details about**  
 275 **the grid acquisition.** We consider a set of 40 different grids (or images). Each grid  
 276 of 20\*50 meters is built with uniform cells of size 0.1\*0.1 meters. Each grid is repre-  
 277 sented as an image in which one pixel corresponds to one cell, the green color shows  
 278 the free cells (usable), the red one shows the occupied cells, while the black repre-  
 279 sents unexplored cells (unknown). The color intensity reflects the certainty degree.  
 280 **For each image (or grid), we generated  $n$  possible trajectories, as represented in**  
 281 **Figure 5.** The shapes of the trajectories are clothoids, that are generated following  
 282 the approach proposed by Mouhagir *et al.*<sup>22</sup>. The number of generated trajectories  
 283 depends on internal states of ego-vehicle (speed and yaw rate). This type of trajec-  
 284 tory called “tentacle” is indeed well suited to local planning. The final aim is then to  
 285 automatically select the best trajectory, or “tentacle”. In the following we consider  
 286  $n = 5$  trajectories. Generating such a finite set of pre-computed trajectories is quite  
 287 common when needing real-time decision for objects having a high velocity (rockets,  
 288 cars, etc.), as it speeds up the decision process. In this case, exploring all solutions



Fig. 4. Experimental vehicle

289 and trajectories may simply be too costly, even with efficient algorithms at hand.

290 Each trajectory is composed of several metagrids, each of them are defined as a  
 291 square encompassing a circle of 3 meters diameter (30\*30 cells). The security distance  
 292 is fixed to 15 meters which is equivalent to 5 metagrids along a trajectory. The  
 293 red circles indicate that the distance of the vehicle to the metagrid is less than the  
 294 security distance. Each cell  $X$  of a metagrid is characterized by a basic belief mass  
 295 assignment (bba)  $m$  on the frame  $\mathcal{X} = \{x, \neg x\}$ , where  $x$  stand for “occupied” and  $\neg x$   
 296 for “free”, that can be easily transformed into lower/upper probabilistic bounds<sup>c</sup>.  
 297 This means that for each cell, we retrieve three masses  $m(\{x\}), m(\{\neg x\}), m(\mathcal{X})$ .  
 298 Each metagrid, initially of size 30\*30, was transformed into metagrids of size 6\*6  
 299 by averaging the masses of adjacent cells.

300 To have access to a ground truth and to be able to compare methods for choosing  
 301 the best trajectory, we have used two human “experts” (in this case, experienced  
 302 drivers from the laboratory) to label the possible trajectories. An example of some  
 303 evaluations given by one expert is given in Table 6. The instructions given to the  
 304 experts were to assign a rank to the proposed trajectories, with the convention  
 305 that 0 means that the trajectory is not at all acceptable (i.e., could lead to an  
 306 accident), that rank 1 is the best rank, and that the same rank can be given to several  
 307 trajectories if the expert is not able to distinguish them in terms of drivability. For  
 308 example, in Table 6, for grid 200, trajectory  $\tau_5$  is declared unacceptable by the  
 309 expert, and  $\tau_3$  is preferred to  $\tau_2$ , which is in turned preferred to  $\tau_1$  and  $\tau_4$ , without  
 310 any preference between these two last trajectories. In image 209,  $\tau_4$  and  $\tau_5$  are  
 311 unacceptable,  $\tau_2$  and  $\tau_3$  are equally preferred, and  $\tau_1$  is in the third position.

To apply our method, we convert the evidential grids (which we simply use as  
 an estimation/learning tool) into imprecise probability grids in the following a way:  
 each cell of a metagrid is characterized by an interval of probability  $[p(x), \bar{p}(x)]$  such  
 that:

$$\underline{p}(x) = m(\{x\}),$$

<sup>c</sup>We consider in this paper only normal bbas so that  $m(\emptyset) = 0$  for any cell of the grid, so we can just treat them as our way to obtain probability bounds, without considering their potential semantic within evidence theory.

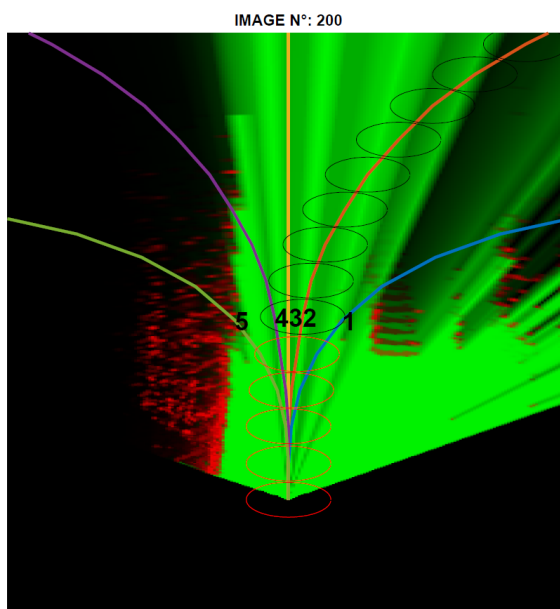


Fig. 5. A credal grid and five possible trajectories

Grid	$\tau_1$	$\tau_2$	$\tau_3$	$\tau_4$	$\tau_5$
200	3	2	1	3	0
209	2	1	1	0	0

Table 6. Example of trajectory evaluation by an expert

and

$$\bar{p}(x) = m(\{x\}) + m(\{x, \neg x\}).$$

312

313 The decision is based on 12 metagrids per trajectory which corresponds in av-  
 314 erage to what can be visually evaluated by the expert. As it is partly outside of  
 315 what can be seen by the Lidar, the first metagrid is not taken into account in the  
 316 decision process. We have fixed arbitrarily the values of the utility function to -5  
 317 for the first four metagrids, and to values linearly spaced between 10 and 70 for the  
 318 following ones.

319 As a baseline to compare our method, we also convert the evidential grids into  
 320 binary grids and use them as follows:

321 (1) A cell is occupied if the middle of the interval  $[\underline{p}(x); \bar{p}(x)]$  is greater than

322 0.5.

- 323 (2) A metagrid is free if none of the cells that make it up are occupied,  
 324 (3) If the rank of the first occupied metagrid is less or equal to 4, then the  
 325 trajectory is declared unacceptable.  
 326 (4) The further away is the first metagrid occupied along a trajectory, the  
 327 better is the trajectory.

328 The reason for choosing such a baseline is that it is very simple (hence easily usable  
 329 by an autonomous car or any mobile robot), but retains very little of our initial, rich  
 330 information. A key question is therefore to check that retaining such a rich (but  
 331 computationally more demanding) information does bring some practical advan-  
 332 tage. The methods are evaluated according to the two tasks performed during the  
 333 decision: deciding if a trajectory is acceptable or not and ordering of the remaining  
 334 trajectories. The methods are thus evaluated according to two different criteria.

For the first task, we use the traditional  $F_\beta$  measure which is defined by:

$$F_\beta = \frac{(1 + \beta^2)TP}{(1 + \beta^2)TP + \beta^2FN + FP},$$

335 where TP stands for the number of true positives, FN for false negatives and FP for  
 336 false positives. In the task of recognizing acceptable trajectories, the false negative  
 337 refer to acceptable trajectories which are wrongly classified as unacceptable, whereas  
 338 the false positives refer to unacceptable trajectories which are wrongly declared as  
 339 acceptable. The last error being much more serious and dangerous for a driver than  
 340 the former, we use a value of  $\beta = 0.5$  which penalizes more the false positives than  
 341 the false negatives. Two possible rules have been explored in the experiments:

- 342 • Rule 1. trajectory  $\tau_j$  is classified as acceptable if  $\underline{\mathbb{E}}(\tau_j) > 0$  ;  
 343 • Rule 2. trajectory  $\tau_j$  is classified as acceptable if  $\overline{\mathbb{E}}(\tau_j) > 0$  .

344 For the second task, we need a criterion to judge the relevance of the trajec-  
 345 tory ordering. As explained before, the order relations on the set of trajectories  
 346 obtained using our method are either total preorders (orders 3 and 4) or partial  
 347 orders (orders 1 and 2). On the other hand, the evaluations of the experts, express-  
 348 ing their preference about the trajectories with possible indifference, lead to total  
 349 preorders. **A first way to compare the results is to use** a general measure to evaluate  
 350 the discrepancy between preorders, as partial or total orders are special cases of  
 351 preorders. For this purpose, we use a distance measure between preorders proposed  
 352 in <sup>15</sup>. Let us use the following notations:  $\tau_j \succ \tau_k$  denotes the fact that  $\tau_j$  is strictly  
 353 preferred to  $\tau_k$ ,  $\tau_j \approx \tau_k$  that  $\tau_j$  and  $\tau_k$  are indifferent, and  $\tau_j ? \tau_k$  means that  $\tau_j$  and  
 354  $\tau_k$  are incomparable. The distance between two preorders is defined as the sum of  
 355 the distances between all pairs of trajectories, using the values indicated in Table  
 356 7. These values have been arbitrarily chosen in a set of values respecting axiomatic  
 357 and common sense rules.

358 **For Orders 3 and 4, and for the binary method, as they lead to total preorders,**  
 359 **there is a second way of comparing the results: if indecision is not allowed, each**

	$\tau_j \approx \tau_k$	$\tau_j \succ \tau_k$	$\tau_j \sim \tau_k$	$\tau_k \succ \tau_j$
$\tau_j \approx \tau_k$	0	1	4/3	1
$\tau_j \succ \tau_k$	1	0	4/3	5/3
$\tau_j \sim \tau_k$	4/3	4/3	0	4/3
$\tau_k \succ \tau_j$	1	5/3	4/3	0

Table 7. Values of the distance between two preorders

360 method can be forced to select a best trajectory i.e. a trajectory which is ranked  
 361 first (there may be multiple ones in case of indifference) in the preorder. We propose  
 362 in this case to evaluate each method by an accuracy score computed as follows. Let  
 363  $\hat{\tau}_i$  denote the prediction of a method. For each predicted trajectory, the individual  
 364 accuracy is obtained as:

$$acc_i = \begin{cases} \frac{1}{r_i} & \text{if } \hat{\tau}_i \text{ is acceptable for the expert} \\ 0 & \text{else,} \end{cases}$$

365 where  $r_i$  denote the rank of  $\hat{\tau}_i$  in the expert order. The total accuracy  $Acc$  is obtained  
 366 as the mean of the individual accuracies.

367 It must be noticed that, in contrast with the IP method, the binary method  
 368 often provides several trajectories equally ranked at the first place because the  
 369 order depends only on the number of metagrids on the trajectories supposed to  
 370 be free. In case of a tie, we propose thus two strategies: either to retain the best  
 371 rank among the set of best trajectories (which gives clearly an advantage to the  
 372 binary method) or to break the tie by randomly choosing a trajectory in the best  
 373 trajectories, which seems fairer for the comparison, as in principle we do not have  
 374 access to the expert “ground truth”.

375 Results are presented in Table 8,9, and 10. In Table 8 is presented the value of  
 376  $F_\beta$  averaged on the 40 images for the two rules and the binary approach. We can see  
 377 that the best score for  $F_\beta$  is achieved using Rule 2. This rule consists in eliminating  
 378 trajectories for which the utility is for sure less than zero. This confirms that using  
 379 negative utilities for the first metagrids is a useful feature, as it is an efficient filter.  
 380 Note that results of our method and the binary one are not very different. As future  
 381 works, we could try to optimize/change utilities so that they provide optimal  $F_\beta$   
 382 on a training set, yet this would significantly increase the method complexity, with  
 383 an unclear benefit.

384 Much more differences are found in the second task. Table 9 presents the results  
 385 obtained for the ordering task (after using Rule 2 for selecting the acceptable trajec-  
 386 tories). It gives the average distance between the expert orders and the automatic  
 387 orders. It can be seen that, for both experts, the order obtained using Order 4 leads  
 388 to a smallest distance with respect to the expert order and that it outperforms the

389 simple binary method. Interestingly, this indicates that once the undesirable (unac-  
 390 ceptable) trajectories have been removed from the possible choices, the optimistic  
 391 behaviour seems to be in very good accordance with the observed expert behaviours.  
 392 Order 1 seem to be too conservative, as could be reasonably expected, while orders  
 393 2 and 3 end up in similar distances. It should however be emphasized that we are  
 394 systematically comparing all orders to total preorders, meaning that it is hard, from  
 395 our experimental setting, to assess the utility of the incomparabilities induced by  
 396 orders 1 and 2, in particular the refine an imprecise selection. Exploring this aspect  
 397 is the matter of future methodological and experimental work, for example to allow  
 398 experts to express such incomparabilities.

399 In Table 10, we compare the results of the task consisting in choosing only one  
 400 trajectory. We report three results for the simple binary method: the maximum ac-  
 401 curacy which corresponds to breaking ties in choosing systematically the trajectory  
 402 associated to the best result (the one ranked highest by the expert), the minimum  
 403 accuracy which corresponds to choosing systematically the worst result and, finally,  
 404 a random guess among the possible best trajectories. Without surprise, the results  
 405 of the binary method in which the maximum accuracy is retained gives the best  
 406 results but it may be seen that Order 4 outperforms the random guess and pro-  
 407 vides results which are close to maximum accuracy. Another thing is that the span  
 408 between the minimum and maximum accuracies can be quite large, thus indicating  
 409 that a refined modelling of uncertainty may be helpful ot better discriminate the  
 410 trajectories.

Expert	$F_\beta$ (rule 1)	$F_\beta$ (rule 2)	$F_\beta$ (binary)
1	0,8934	<b>0,9118</b>	0,9034
2	0,8768	<b>0,9460</b>	0,9429

Table 8. Classification unacceptable/acceptable: average  $F_\beta$  for the three methods

Expert	Order 1	Order 2	Order 3	Order 4	Binary
1	9,7	6,31	6,85	<b>2,81</b>	6,84
2	9,86	6,4	6,2	<b>3,61</b>	7,32

Table 9. Ordering results: mean distance between the expert and the proposed order

Expert	Order 3	Order 4	Binary (min)	Binary (random)	Binary (max)
1	0,6995	0.9125	0,5372	0.7855	0.9251
2	0,6663	0,8792	0,4422	0,6446	0,9625

Table 10. **Best trajectory results: accuracy of the methods**

## 411 5. Conclusion

412 In this article, we have introduced the notion of credal occupancy grids, that to  
413 the best of our knowledge was never studied before. The advantage of such grids is  
414 their flexibility in terms of uncertainty modelling, as well as their consistency with  
415 traditional probabilistic models, that they extend. To show how such grids could  
416 be used in practice, we have proposed a model to make cautious inferences for  
417 robot trajectory planning from uncertain information provided in the form of such  
418 credal occupancy grids. The proposed model allows one to accurately represent  
419 the available information while controlling the computational complexity by the  
420 use of particular events of the decision space. In particular, the proposed method is  
421 relatively simple and makes very clear assumptions about the uncertainty modelling  
422 and reasoning parts, as well as about the decision process. This has in our opinion  
423 two key advantages: it makes the whole method more acceptable and explainable to  
424 third parties, and it is easy to revisit, question, or modify the different components  
425 of the method.

426 We have illustrated the behaviour of the proposed approach using simple ex-  
427 amples, and have tested it using real data, in the context of autonomous vehicles,  
428 comparing it favourably with a simple baseline approach. These first results indeed  
429 show that the approach is interesting thanks to a greater flexibility in the decision  
430 process. To completely validate the approach, a greater number of experts and a  
431 larger number of grids and situations in the experimental part would be needed.  
432 Future research could include also the automatic learning of the utility function  
433 and the integration of other criteria in the decision process like the similarity to a  
434 reference trajectory.

435 We also plan to augment the expressivity of the method, for example by bet-  
436 ter differentiating cell occupancy information, distinguishing between completely  
437 usable cells, not-fully usable cells (e.g. for vehicles, side-walks, emergency lines)  
438 and completely unusable cells (with obstacles). Such distinctions could be impor-  
439 tant elements to deal with complex situations possibly leading to accidents, but  
440 would increase the method complexity. Another important avenue of research is  
441 how dependence information could be embedded in the inferences. Indeed, so far  
442 we have considered the cells and their uncertainty model to be independent, which  
443 is an acceptable simplifying assumption in first approximation, but is not true in  
444 general.

445 Finally, while we considered here the problem of making real-time inferences  
446 on a pre-selected number of trajectories by using DNF Boolean formulas, it would  
447 be interesting to investigate other ways to handle either the planning problem or  
448 to obtain the normal forms of the formulas, that could prove useful in those cases  
449 where computational time should remain acceptable but is not constrained by real-  
450 time and by physical constraints induced by velocity and limiting the number of  
451 trajectories one could consider (e.g., slow moving robots in risky environments). For  
452 instance, one could consider finding an arbitrary optimal sequence of cells as a tra-  
453 jectory from point  $A$  to point  $B$ , taking inspiration from recent works on inferences  
454 in HMM<sup>8</sup> or from works extending PODMP<sup>14</sup> to the imprecise framework. Con-  
455 cerning Boolean formulas, we mainly used DNF formulas and mentioned BDDs as  
456 another way to obtain compact representations of Boolean functions, but it would  
457 interesting to make a general exploration of the interplay between different ways of  
458 compiling semantically equivalent Boolean formulas<sup>7</sup> and their corresponding com-  
459 putational complexity, notably those compilations that have an efficient imprecise  
460 counterpart such as the recent Sentential Decision Diagrams<sup>18</sup>.

#### 461 Acknowledgements

462 This work was carried out in the framework of Equipex ROBOTEX (ANR-10-  
463 EQPX-44-01) and Labex MS2T (ANR-11-IDEX-0004-02), which were funded by  
464 the French Government, through the program “Investments for the future” managed  
465 by the National Agency for Research.

- 466 1. F. Aguirre, S. Destercke, D. Dubois, M. Sallak, and C. Jacob. Inclusion–exclusion prin-  
467 ciple for belief functions. *International Journal of Approximate Reasoning*, 55(8):1708–  
468 1727, 2014.
- 469 2. Thomas Augustin, Frank PA Coolen, Gert De Cooman, and Matthias CM Troffaes.  
470 *Introduction to imprecise probabilities*. John Wiley & Sons, 2014.
- 471 3. Endre Boros and Peter L Hammer. Pseudo-boolean optimization. *Discrete applied*  
472 *mathematics*, 123(1-3):155–225, 2002.
- 473 4. R. E. Bryant. Symbolic boolean manipulation with ordered binary-decision diagrams.  
474 *ACM Computing Surveys (CSUR)*, 24(3):293–318, 1992.
- 475 5. Davide Ciucci. Orthopairs: A simple and widely used way to model uncertainty. *Fun-*  
476 *damenta Informaticae*, 108(3-4):287–304, 2011.
- 477 6. C. Coue, C. Pradalier, C. Laugier, T. Fraichard, and P. Bessiere. Bayesian occupancy  
478 filtering for multitarget tracking: an automotive application. *The International Jour-*  
479 *nal of Robotics Research*, 25(1), pp. 19-30, 2006.
- 480 7. Adnan Darwiche and Pierre Marquis. A knowledge compilation map. *Journal of Ar-*  
481 *tificial Intelligence Research*, 17:229–264, 2002.
- 482 8. Jasper De Bock and Gert De Cooman. An efficient algorithm for estimating state  
483 sequences in imprecise hidden markov models. *Journal of Artificial Intelligence Re-*  
484 *search*, 50:189–233, 2014.
- 485 9. L.M. de Campos, J.F. Huete, and S. Moral. Probability intervals: a tool for uncer-  
486 tain reasoning. *International Journal of Uncertainty, Fuzziness and Knowledge-Based*  
487 *Systems*, 2:167–196, 1994.

- 488 10. S. Destercke and D Dubois. Special cases. In *Introduction to Imprecise Probabilities*,  
489 chapter 4, pages 79–92. John Wiley & Sons, Ltd, 2014.
- 490 11. P.C. Fishburn. Exceptional paper-lexicographic orders, utilities and decision rules: A  
491 survey. *Management science*, 20(11):1442–1471, 1974.
- 492 12. J. Fortin, D. Dubois, and H. Fargier. Gradual numbers and their application to fuzzy  
493 interval analysis. *IEEE Transactions on Fuzzy Systems*, 16(2):388–402, 2008.
- 494 13. Wolfgang Gatterbauer and Dan Suciu. Oblivious bounds on the probability of boolean  
495 functions. *ACM Transactions on Database Systems (TODS)*, 39(1):1–34, 2014.
- 496 14. Hideaki Itoh and Kiyohiko Nakamura. Partially observable markov decision processes  
497 with imprecise parameters. *Artificial Intelligence*, 171(8-9):453–490, 2007.
- 498 15. K. Jabeur, J.-M. Martel, and S. B. Khélifa. A distance-based collective preorder integrat-  
499 ing the relative importance of the group’s members. *Group Decision and Negoti-*  
500 *ation*, 13(4):327–349, 2004.
- 501 16. Paul A Jensen and Mandell Bellmore. An algorithm to determine the reliability of a  
502 complex system. *IEEE Transactions on reliability*, 18(4):169–174, 1969.
- 503 17. Heudiasyc Laboratory. Plateforme PACPUS. <https://pacpus.hds.utc.fr/>.  
504 Accessed:2019-11-04.
- 505 18. Lilith Mattei, Alessandro Antonucci, Denis Deratani Mauá, Alessandro Facchini, and  
506 Julissa Villanueva Llerena. Tractable inference in credal sentential decision diagrams.  
507 *International Journal of Approximate Reasoning*, 125:26–48, 2020.
- 508 19. J. Moras, C. Cherfaoui, and P. Bonnifait. Moving objects detection by conflict analysis  
509 in evidential grids. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 1122–1127,  
510 Baden-Baden, Germany, June 2011.
- 511 20. J. Moras, V. Cherfaoui, and P. Bonnifait. Credibilist occupancy grids for vehicle per-  
512 ception in dynamic environments. In *IEEE International Conference on Robotics and*  
513 *Automation (ICRA)*, pages 84 – 89, Shangai, China, May 2011.
- 514 21. H. Moravec and A. Elfes. High resolution maps from wide angle sonar. In *IEEE*  
515 *International Conference on Robotics and Automation (ICRA)*, volume 2, pages 116  
516 – 121, St Louis, USA, March 1985.
- 517 22. Hafida Mouhagir, Reine Talj, Véronique Cherfaoui, François Aioun, and Franck Guille-  
518 mard. Evidential-Based Approach for Trajectory Planning With Tentacles, for Au-  
519 tonomous Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, pages  
520 1–12, 2019.
- 521 23. S. Noykov and C. Roumenin. Occupancy grids building by sonar and mobile robot.  
522 *Robotics and Autonomous Systems*, 55(2):162–175, 2007.
- 523 24. G. Oriolo, G. Uliviy, and M. Vendittelli. Fuzzy maps: A new tool for mobile robot  
524 perception and planning. *Journal of Robotic Systems*, 14(3):179–197, 1997.
- 525 25. Guillermo Owen. Multilinear extensions of games. *Management Science*, 18(5-part-  
526 2):64–79, 1972.
- 527 26. D. Pagac, E. Nebot, and H. Durrant-Whyte. Evidential approach to map-building for  
528 autonomous vehicles. *IEEE Transactions on Robotics and Automation*, 14(4):623–629,  
529 1998.
- 530 27. Chandima Dedduwa Pathirana, Keigo Watanabe, and Kiyotaka Izumi. T-s fuzzy  
531 model adopted slam algorithm with linear programming based data association for  
532 mobile robots. *Soft Computing*, 14(4):345, 2010.
- 533 28. A. Saffiotti. The uses of fuzzy logic in autonomous robot navigation. *Soft Computing*,  
534 1(4):180–197, 1997.
- 535 29. P. Smets. Decision making in the TBM: the necessity of the pignistic transformation.  
536 *International Journal of Approximate Reasoning*, 38:133–147, 2005.
- 537 30. G. Tanzmeister and D. Wollherr. Evidential grid-based tracking and mapping. *IEEE*

- 538        *Transactions on Intelligent Transportation Systems*, 18(6):1454–1467, 2017.
- 539 31. M.C.M. Troffaes. Decision making under uncertainty using imprecise probabilities.
- 540        *International Journal of Approximate Reasoning*, 45:17–29, 2007.
- 541 32. P. Walley. *Statistical reasoning with imprecise probabilities*. Chapman and Hall, New
- 542        York, 1991.
- 543 33. T. Yang and V. Aitken. Evidential mapping for mobile robots with range sensors.
- 544        *IEEE Transactions on Instrumentation and Measurement*, 55(4):1422–1429, 2006.