Hierarchical path planning on probabilistically labeled polygons

Emmanuel Piat*
Heudiasyc UMR CNRS 6599
BP 20 529
60205 Compiègne Cedex
France

Simon Lacroix†
LAAS CNRS
7, av. du Colonel Roche
31077 Toulouse Cedex
France

Abstract

In this paper, we consider the case of a robot that must find ways in an initially unknown and often complex cross-country environment. Our approach relies on a particular model of the environment, built from 3D data : the environment is represented by a hierarchical polygonal structure, in which terrain classes probabilities are updated as the robot moves. We describe how to define traversability costs within such a structure, taking into account the label probabilities, and present how a shortest path search can produce a sequence of polygonal cells that must be crossed.

1 Introduction

One of the main difficulty of cross-country autonomous robotics comes from the fact that outdoor environments can hardly be faithfully modeled: such environments are not intrinsically structured, as compared to indoor environments, where simple geometric primitives match the reality. As a consequence, any representation based only on geometric primitives is difficult to build and to maintain.

We consider in this paper the problem of maintaining a *global* qualitative model that describes the environment as a set of polygonal cells, for which terrain classes probabilities are estimated, and the way to use such a model in order to plan minimum cost paths. Our planner fits in the family of "weighted regions planners" ([8, 10]), but we rely on a more complex hierarchical structure instead of a partition of the perceived regions: we will see that this allows to explicitly represent and exploit *all* the information gathered by the robot during the modeling phase.

The paper is organized as follows : the next section briefly presents a 3D data clustering procedure that leads to a description of the terrain in polygonal cells, for which a probability to cover a given terrain class is estimated. Section 3 describes how the cells probabilities extracted from images acquired at various positions are combined to produce a global coherent model. Section 4 is devoted to the path planner : we define an hierarchical graph representation based upon the aggregation of polygonal cells, then present how traversability costs can be estimated on the basis of the terrain class probabilities, and finally explain how a minimum path search lead to the definition of the navigation area.

2 Terrain classification

Our terrain modeling procedure relies on a specific discretisation of the perceived area, that defines a *cell image*. Attributes are computed for each cell, and are used to label the cells in term of *navigation classes*, thanks to a supervised Bayesian classifier. The discretisation being imposed, this procedure is rather an *identification* process, and do not require a segmentation of the 3D data, a quite difficult problem on complex terrains (more details can be found in [3]).

2.1 Cells definition

The sensors that produce 3D points, be it a laser range finder or a stereo-vision correlation algorithm, generally have a constant scanning rate within the *sensor frame*. But when perceiving a ground plane with such sensors, the resolution of the perceived points decreases dramatically with their distance to the sensor. This is a fundamental point one can not ignore when processing such data : it lead us to choose the discretisation of the perceived zone presented in figure 1, instead of a Cartesian one such as in [5]. This discretisation corresponds to the central projection on a virtual horizontal ground of a regular (Cartesian) and convex discretisation in the sensor frame. The fundamental property of this discretisation is the "conserva-

[*] E-mail: Emmanuel.Piat@hds.utc.fr

[†] E-mail: Simon.Lacroix@laas.fr

tion of density" : if a perfect flat ground corresponding to the reference plane is perceived, the number of points that belong to a cell - *i.e.* whose vertical projection coordinates are bounded by the cell's border - is equal to a constant *nominal density*, defined by the discretisation rates. On the other hand, a cell covering an obstacle area contains much more points than the nominal density. The number of points contained in a cell becomes obviously an important feature to determine the nature of the perceived zone covered by this cell. Other features are used to identify a cell : the elevation standard deviation and the maximum elevation difference computed on the cell's points give an information on the "flatness" of the cell ; the mean normal vector and the variances on its coordinate are useful to determine if the cell covers a regular slope or has an irregular surface.

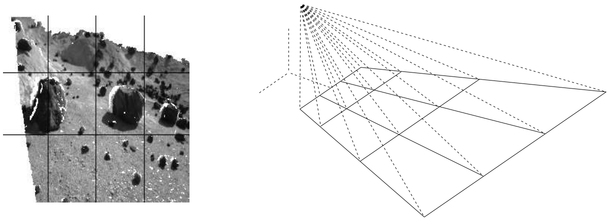

Figure 1: *The regular Cartesian discretisation in the sensor frame, and its projection on the ground*

2.2 Cell classification

A supervised Bayesian classification procedure is used to label each cell: during an off-line learning phase, a human prototyped a set of cell images. On line, once 3D data are acquired, the cells image is defined by the discretisation, and the value x of the feature vector x^c is computed for each cell C. The Bayes theorem is then applied to determine the partial probabilities $P(w_i^c|x^c = x)$ for a cell to correspond to each of the M terrain classes $\{\omega_1, \ldots, \omega_M\}$:

$$P(w_i^c|X^c) = \frac{f_{x^c}(x|w_i^c)P(w_i^c)}{\sum_{j=1}^{M} f_{x^c}(x|w_j^c)P(w_j^c)} \qquad (1)$$

where X^c stands for "$x^c = x$", w_i^c stands for "$\omega^c = \omega_i$", $f_{x^c}(x|w_i^c)$ is the value at x of the probability density function f of the random feature vector x^c *for all cells C labeled* ω_i and $P(w_i^c)$ the *a priori* probability that the class of cell C is ω_i. The pdf's $f(x|w_i^c)$ can hardly be analytically expressed : their value at the point x is computed using a nearest neighbor technique within the feature space filled with the prototyped cells feature values. Figure 2 presents a classification result considering three terrain classes $\{$*Flat - with an admissible slope, Uneven, Obstacle*$\}$.

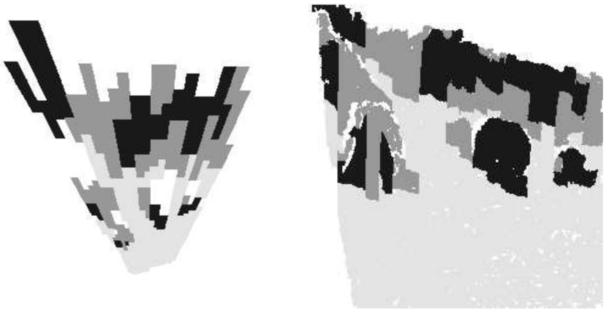

Figure 2: *Classification of the stereo image of figure 1 : perceived area (left) and reprojection in the sensor frame (right). From clear to dark: Unknown, Flat, Uneven, Obstacle*

3 Global model updating

3.1 The problem

As the robot moves, a new cell image is built each time 3D data are acquired. Provided a good estimation of the robot position is known, one can build a global model, aggregating the various cell images. The fusion problem comes here to estimate the partial probabilities $P(w_i)$ of the terrain regions where cells overlap.

Consider two overlapping cells labeled a and b, defined in two different cell images (figure 5). Let x_a (resp. x_b) the value of the feature vector x^{c_a} (resp. x^{c_b}) extracted from cell C_a (resp. C_b). Let ω^{c_a} and ω^{c_b} the classes of cells C_a and C_b, and $\omega^{c_{a,b}}$ the class of the intersection $C_{a,b} = C_a \cap C_b$. One could try to label $C_{a,b}$ conditionnaly to the knowledge of a feature vector $x^{c_{a,b}}$ extracted from this cell. However, this would require to memorize all the 3D data points acquired at different times, and to build a really huge number of prototypes of intersecting cells, which is impossible in practice. One could also use Bayesian fusion:

$$\begin{aligned} & P(w_i^{c_{a,b}}|X^{c_a}X^{c_b}) \\ & \quad = \frac{f_{x^{c_a}}(x_a|w_i^{c_{a,b}}X^{c_b})P(w_i^{c_{a,b}}|X^{c_b})}{\sum_{j=1}^{M} f_{x^{c_a}}(x_1|w_j^{c_{a,b}}X^{c_b})P(w_j^{c_{a,b}}|X^{c_b})} \qquad (2) \end{aligned}$$

But again, the computation of terms $P(w_i^{c_{a,b}}|X^{c_b})$ and $f_{x^{c_a}}(x_a|w_i^{c_{a,b}}X^{c_b})$ requires a huge number of prototypes (for instance, $P(w_i^{c_{a,b}}|X^{c_b})$ is the probability that an overlapping of cell C_b is labeled ω_i when the feature vector extracted from C_b is $x^{c_b} = x_b$).

3.2 Our approach

In order to solve the fusion problem, we will first determine $P(w_i^{c_{a,b}}|X^{c_a})$ (or $P(w_i^{c_{a,b}}|X^{c_b})$) thanks to a model, and then use a simplifying hypothesis in order to easily compute $P(w_i^{c_{a,b}}|X^{c_a}X^{c_b})$.

$$P(w_i^{c_{a,b}}|X^{c_a}) = P(w_i^{c_{a,b}}(w_1^{c_a}+\cdots+w_M^{c_a})|X^{c_a}) \qquad (3)$$

$$= \sum_{j=1}^{M} P(w_j^{c_a}|X^{c_a})P(w_i^{c_{a,b}}|w_j^{c_a}X^{c_a}) \qquad (4)$$

To estimate $P(w_i^{c_{a,b}}|w_j^{c_a}X^{c_a})$ in (4), we use a model based on surfaces s^{c_a} and $s^{c_{a,b}}$ of cells C_a and $C_{a,b}$. Obviously, when labels i and j are equal, it comes:

$$\begin{cases} P(w_i^{c_{a,b}}|w_i^{c_a}X^{c_a}) = 1 & \text{if } s^{c_{a,b}} = s^{c_a} \\ P(w_i^{c_{a,b}}|w_i^{c_a}X^{c_a}) \to \frac{1}{M} & \text{if } s^{c_{a,b}} \to 0 \end{cases} \qquad (5)$$

The model we choose assumes that as the overlapping surface increases, the probability that both cells C_a and $C_{a,b}$ labels are the same is higher. We therefore estimate $P(w_i^{c_{a,b}}|w_i^{c_a}X^{c_a})$ with a strictly increasing function of $\frac{s^{c_{a,b}}}{s^{c_a}}$:

$$\widehat{P}(w_i^{c_{a,b}}|w_i^{c_a}X^{c_a}) = g\Big(\frac{s^{c_{a,b}}}{s^{c_a}}\Big) \qquad (6)$$

With a linear estimation and with (5) (the simplest model one can make without any more information) it comes:

$$g\Big(\frac{s^{c_{a,b}}}{s^{c_a}}\Big) \triangleq \frac{M-1}{M} \times \frac{s^{c_{a,b}}}{s^{c_a}} + \frac{1}{M} \qquad (7)$$

When $i \neq j$, $\widehat{P}(w_j^{c_{a,b}}|w_i^{c_a}X^{c_a}X^{c_b})$ is supposed to be uniformly distributed over $1-\widehat{P}(w_i^{c_{a,b}}|w_i^{c_a}X^{c_a})$ because the system has no reason to privilege any other possible labels:

$$\widehat{P}(w_j^{c_{a,b}}|w_i^{c_a}X^{c_a}) = \frac{1}{M-1}\Big[1-g\Big(\frac{s^{c_{a,b}}}{s^{c_a}}\Big)\Big]$$

$$= \frac{1}{M}\Big(1-\frac{s^{c_{a,b}}}{s^{c_a}}\Big) \qquad \text{if } i \neq j \qquad (8)$$

and (4) becomes

$$P(w_i^{c_{a,b}}|X^{c_a}) = \frac{s^{c_{a,b}}}{s^{c_a}}\Big(P(w_i^{c_a}|X^{c_a}) - \tfrac{1}{M}\Big) + \tfrac{1}{M} \qquad (9)$$

Symmetrically, the same result is obtained for $P(w_i^{c_{a,b}}|X^{c_b})$. The table array illustrates results (9) with $s^{c_a}=10$, $s^{c_b}=5$ and $s^{c_{a,b}}=4$.

	flat	uneven	obstacle	$\sum$
$P(w_i^{c_a}\|X^{c_a})$	0.700	0.200	0.100	1
$P(w_i^{c_{a,b}}\|X^{c_a})$	0.480	0.280	0.240	1
$P(w_i^{c_b}\|X^{c_b})$	0.500	0.400	0.100	1
$P(w_i^{c_{a,b}}\|X^{c_b})$	0.467	0.387	0.146	1

Now, suppose x^{c_a} and x^{c_b} are independent[1] for all classes of $C_a \cap C_b$:

$$f_{x^{c_a}}(x|w_i^{c_{a,b}}X^{c_b}) = f_{x^{c_a}}(x|w_i^{c_{a,b}}) \quad \forall x, i$$

The conjunctive fusion process of $P(w_i^{c_{a,b}}|X^{c_a})$ and $P(w_i^{c_{a,b}}|X^{c_b})$ obtained with (9) gives [9]:

$$P(w_i^{c_{a,b}}|X^{c_a}X^{c_b}) = \frac{P(w_i^{c_{a,b}}|X^{c_a})P(w_i^{c_{a,b}}|X^{c_b})}{\sum_{j=1}^{M} P(w_j^{c_{a,b}}|X^{c_a})P(w_j^{c_{a,b}}|X^{c_b})} \qquad (10)$$

One can see on the following table, that the probability of the $C_{a,b}$ flat label is smaller than the C_a one (0.7) because cell $C_{a,b}$ is smaller than C_a. But this probability is also higher than the C_b one (0.5) because it is reinforced by C_a.

	flat	uneven	obstacle	$\sum$
$P(w_i^{c_{a,b}}\|X^{c_a})$	0.480	0.280	0.240	1
$P(w_i^{c_{a,b}}\|X^{c_b})$	0.467	0.387	0.146	1
$P(w_i^{c_{a,b}}\|X^{c_a}X^{c_b})$	0.609	0.295	0.096	1

4 The planning process

4.1 Problem statement

The planning algorithm has to determine, in some valued graph $\mathcal{G}$, a path $\mathcal{P}$ of connected cells from some starting cell C_s to some arrival cell C_a such that the cost summation on the arcs is minimal. Let $\mathcal{C}_{global}$ the whole surface defined by all observed mother cells C_i:

$$\mathcal{C}_{global} = C_1 \cup C_2 \cup \cdots \cup C_n \qquad (11)$$

Two approaches are possible. First, $\mathcal{C}_{global}$ can be partionned with non-overlapping cells. For instance, with mother cells C_a and C_b on figure 5, the partition will be $\mathcal{C}_{global} = \{C_a - C_{a,b}\ ;\ C_{a,b}\ ;\ C_b - C_{a,b}\}$. Such an approach has been exploited yet in [6, 2, 4] when cells label are known. If labels are unknown, the main problem is that the label distribution $\{P(w_i^{c_a - c_{a,b}}|X^{c_a})\}_i$ defined over $C_a - C_{a,b}$ thanks to our model (9) may be much *less informative* than the distribution $\{P(w_i^{c_a}|X^{c_a})\}_i$ defined over the entire mother cell C_a. The second possible approach we propose here consists in using an overlapping (and not a partition) of $\mathcal{C}_{global}$ built with mother cells and cells

[1] This means that an observed value of x^{c_b} give no information on any feature vector x^{c_a} extracted from any cell C_a such that $C_a \cap C_b$ is labeled ω_i. Smaller is the overlapping compared to C_a and better is this hypothesis.

included in them : $\mathcal{C}_{global} = \{C_a \; ; \; C_{a,b} \; ; \; C_b\}$. With such an approach, it is possible to build a *hierarchical* path planning description [7, 1, 11].

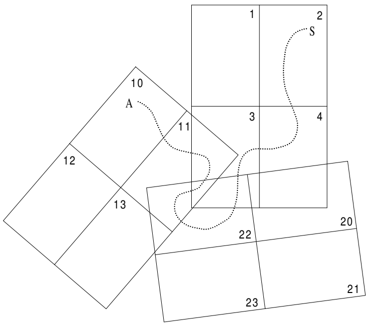

Figure 3: *Three perceived zones divided in 12 cells*

4.2 Definition of the states graph

Our path planning method is based on an exploration of a states graph $\mathcal{G}$ in which nodes are the cells previously defined and arcs represent the topological relations "is included in" and "is adjacent to". This states graph can be represented with a layered structure : the top layer consists of mother cells, the second layer consists of cells obtained by intersection of two mother cells, the third layer consists of cells obtained by intersection of three mother cells, and so on. Connected cells in a same layer are adjacent cells. Figure 4 shows the states graph corresponding to the three perceived areas of figure 3.

4.3 Cost of traversability

The cost of traversability to move from a cell C_a towards a cell C_b when C_a and C_b are connected in $\mathcal{G}$ depends on the physical nature of these both cells. For instance, moving from a flat cell towards another flat one is less expansive than moving from a flat cell towards a uneven one. So this cost is (knowing cells C_a and C_b labels) $\mathcal{C}(C_a \to C_b | w_i^{c_a} w_j^{c_b})$ and, as real cells labels are unknown, this cost is associated to the join probability $P(w_i^{c_a} w_j^{c_b} | X^{c_a} X^{c_b})$. Thus, the average of the cost of traversability which depends on random variables ω^{c_a} and ω^{c_b} is:

$$\mathcal{C}(C_a \to C_b) = E\Big[\mathcal{C}(C_a \to C_b | \omega^{c_a} \omega^{c_b})\Big] \tag{12}$$
$$= \sum_{i=1}^{M} \sum_{j=1}^{M} \mathcal{C}(C_a \to C_b | w_i^{c_a} w_j^{c_b}) P(w_i^{c_a} w_j^{c_b} | X^{c_a} X^{c_b})$$

The computation of $P(w_i^{c_a} w_j^{c_b} | X^{c_a} X^{c_b})$ is developed in the two following sections for adjacent and overlapping cells.

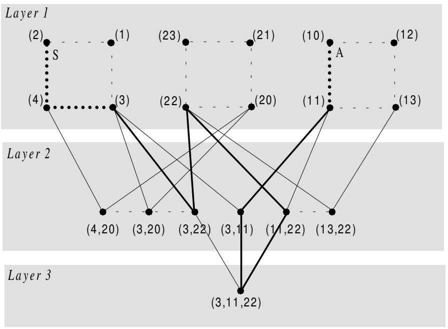

Figure 4: *States graph of perceived zones defined in figure 3*

4.3.1 Adjacent cells

When cells C_a and C_b are adjacent, their labels are independent because features are extracted on different areas. Thus:

$$\begin{aligned} & P(w_i^{c_a} w_j^{c_b} | X^{c_a} X^{c_b}) \\ & \quad = P(w_i^{c_a} | w_j^{c_b} X^{c_a} X^{c_b}) P(w_j^{c_b} | X^{c_a} X^{c_b}) \\ & \quad = P(w_i^{c_a} | X^{c_a} X^{c_b}) P(w_j^{c_b} | X^{c_a} X^{c_b}) \\ & \quad = P(w_i^{c_a} | X^{c_a}) P(w_j^{c_b} | X^{c_b}) \end{aligned} \tag{13}$$

where $P(w_i^{c_a} | X^{c_a})$ and $P(w_j^{c_b} | X^{c_b})$ are known.

4.3.2 Overlapping cells

When C_a and C_b overlaps, the system *necessarily* crosses area $C_{a,b} = C_a \cap C_b$ when moving from C_a to C_b (figure 5). So there is no arc directly connected between (a) and (b) in the states graph. Arcs are between nodes (a) and (a, b) and between nodes (b) and (a, b). Therefore, studied cost is $\mathcal{C}(C_a \to C_{a,b})$.

Fusion process on the top layer : When C_a and C_b are mother cells, one only knows the feature vectors X^{c_a} and X^{c_b}, and not $X^{c_{a,b}}$. So following (12), studied costs $\mathcal{C}(C_a \to C_{a,b})$ or $\mathcal{C}(C_{a,b} \to C_a)$ depends on:

$$\begin{aligned} & P(w_i^{c_a} w_j^{c_{a,b}} | X^{c_a} X^{c_b}) \\ & \quad = P(w_j^{c_{a,b}} | w_i^{c_a} X^{c_a} X^{c_b}) P(w_i^{c_a} | X^{c_a} X^{c_b}) \\ & \quad = P(w_j^{c_{a,b}} | w_i^{c_a} X^{c_a} X^{c_b}) P(w_i^{c_a} | X^{c_a}) \end{aligned}$$

and like for (10) independent conjunctive fusion process of $P(w_j^{c_{a,b}} | w_i^{c_a} X^{c_a})$ and $P(w_j^{c_{a,b}} | X^{c_b})$ (obtained

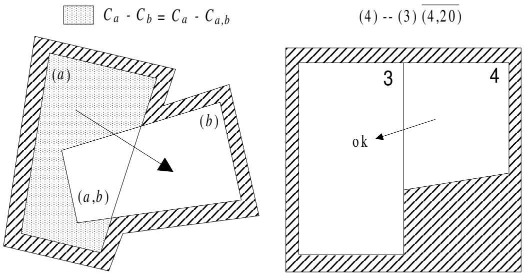

Figure 5: *Overlapping cells and avoided areas*

with (6) and (9)) gives [9]:

$$
\begin{aligned}
&P(w_j^{c_{a,b}}|w_i^{c_a}X^{c_a}X^{c_b}) \\
&\quad = \frac{P(w_j^{c_{a,b}}|w_i^{c_a}X^{c_a})P(w_j^{c_{a,b}}|X^{c_b})}{\sum_{k=1}^{M} P(w_k^{c_{a,b}}|w_i^{c_a}X^{c_a})P(w_k^{c_{a,b}}|X^{c_b})}
\end{aligned} \tag{14}
$$

Fusion process on lower layers : The Lower is the layer and more complicated is the fusion process. Nevertheless, it is always possible to compute the probabilities obtained thanks to relations (6) (9) (10). For instance, on the third layer, a generic cost $\mathcal{C}(C_{a,b} \rightarrow C_{a,b,c})$ depends on the probability $P(w_i^{c_{a,b}}w_j^{c_{a,b,c}}|X^{c_a}X^{c_b}X^{c_c})$ which can be developed in terms obtained thanks to a conjunctive fusion process of $P(w_i^{c_{3,11,22}}|X^{c_{22}})$ given by (9) and $P(w_i^{c_{3,11,22}}|w_j^{c_{3,11}}X^{c_3}X^{c_{11}})$ given by (6).

4.3.3 Graph valuation

The last term to specify in (12) is the cost $\mathcal{C}(C_a \rightarrow C_b|w_i^{c_a}w_j^{c_b})$ defined when cells labels are known. This cost depends on the nature of area included in cell C_a that the robot has to cross in order to reach C_b and the interest of reached cell C_b. These both points may be explained with the following formula:

$$
\begin{aligned}
&\mathcal{C}(C_a \rightarrow C_b|w_i^{c_a}w_j^{c_b}) = \\
&\qquad \delta_1 \underbrace{N(C_a, C_b|w_i^{c_a})}_{1^{\text{st}}\text{ point}} + \delta_2 \underbrace{D(C_b|w_i^{c_a}w_j^{c_b})}_{2^{\text{nd}}\text{ point}}
\end{aligned} \tag{15}
$$

For instance $D(C_b|w_i^{c_a}w_j^{c_b})$ can penalizes displacement towards undulating areas and prevent move towards obstacle areas. It can also enable the system to define a path crossing some specific areas. An example of possible cost N is detailed in section 4.6.

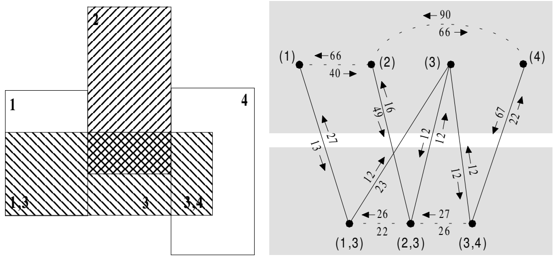

Figure 6: *Basic cells configuration and graph $\mathcal{G}$ associated*

4.4 Planning algorithm

The algorithm we used to explore $\mathcal{G}$ is $\mathcal{A}^*$ which requires the estimation of an heuristic function $h(C_u, C_a)$, that minimizes the cost $h^*(C_u, C_a)$ of all paths in $\mathcal{G}$ between C_u and the goal cell C_a. Generally, heuristic functions used are based on distances in $\mathcal{C}_{global}$. As we work on an overlapping of $\mathcal{C}_{global}$ instead of a partition, the distance used must be defined for any king of cells and not only for adjacent ones like in [2]. The distance we used is the Hausdorff distance h_∞. If A and B are two subsets of $\mathbb{R}^n$

$$h_\infty(A, B) \triangleq \max\{h_\infty^0(A, B), h_\infty^0(B, A)\} \tag{16}$$

where $h_\infty^0(A, B) \triangleq \inf\{r \in \mathbb{R}^+|A \subset B + r\mathcal{U}\}$ and $\mathcal{U}$ is the unit sphere of $\mathbb{R}^n$ with $\mathcal{L}_\infty$ distance

$$\mathcal{L}_\infty(x, y) \triangleq \max_{i \in \{1,\ldots,n\}} |y_i - x_i|$$

4.5 Navigation areas specification

Finally, translation of some path in the hierarchical graph $\mathcal{G}$ into some crossing areas obey these two rules:

Rule 1 *When moving from cell j to cell i, the robot must avoid all connected cells to j in lower layers.*

Rule 2 *When moving from cell j to cell i, the robot must leave cell j if $C_j \subset C_i$.*

For instance, if the planning process gives the path $\mathcal{P}$: $(2) - (4) - (3) - (3, 22) - (22) - (11, 22) - \ldots$ shown on figure 4, the subpath $(4) - (3)$ means that the robot has to go directly from cell 4 to cell 3 *without crossing the zone* $(4, 20)$ (cf. figure 5) because the subpath will be is this last case: $(2) - (4) - (4, 20) - \ldots$ and not $(2) - (4) - (3) - \ldots$ (cf. figure 4). Therefore, the choice $(4) - (3)$ means implicitly that the robot avoids other choices represented by smaller cells *included* in 4 and thus situated in lower layers. The univoque path corresponding to these rules is represented in dashed line on figure 3.

4.6 Example

Figure 6 shows a basic example of cells configuration to which are attached the following distributions:

	flat	uneven	obstacle	area
$P(w_i^{c_1}\|X^{c_1})$	0.7	0.2	0.1	6
$P(w_i^{c_2}\|X^{c_2})$	0.2	0.8	0	8
$P(w_i^{c_3}\|X^{c_3})$	0.8	0.1	0.1	10
$P(w_i^{c_4}\|X^{c_4})$	0.2	0.7	0.1	8

One has $s^{c_{1,3}} = 4$, $s^{c_{2,3}} = 2$, $s^{c_{3,4}} = 2$. We choose $\delta_1 = 1$ and $\delta_2 = 0$ in (15). Thus the interest of reached cells is not taken into account for the mission. Here, the cost function N only depends on the time required to define and execute a *trajectory* in $\mathcal{C}_{global}$. This time is theoretically the sum of terrain modeling time (from perception to model production) and trajectory planning time. If the robot has to cross C_a in order to reach C_b, we suppose that this time is proportional to the area surface the robot has to model in order to compute an executable trajectory.

$$N(C_a, C_b|w_i^{c_a}) = \alpha(\omega_i)s^{c_a-(c_b\cup c^*)} \qquad (17)$$

where $\alpha(\omega_i)$ is the cost of the modeling-planning process for unitary cell surface with label ω_i and:

$$\alpha(flat) < \alpha(uneven) < \alpha(obstacle)$$

and c^* are the avoided areas then moving from C_a to C_b (cf. rule 1 in section 4.5). With $\alpha(flat) = 1$, $\alpha(uneven) = 10$, $\alpha(obstacle) = 40$ the planner gives the following path if the robot has to move from C_1 to C_4 : $(1) - (1,3) - (3) - (3,4) - 4$. So the robot chooses to avoid the large and uneven area $C_2 - C_{2,3}$ and prefers to cross area $C_3 - (C_{1,3} \cup C_{2,3} \cup C_{3,4})$ where it has a more accurate label knowledge.

5 Conclusion

The hierarchical path planner described in this paper has two main advantages: on one hand, the cost function used does not require the application of any decision function after the cell partial probabilities estimation : all the information gathered by the robot remains fully explicited. On the other hand, the hierarchical structure of the global model we consider enables to control the planning complexity. This structure is also compatible with a multi-resolution discretisation of the environment $\mathcal{C}_{global}$. For instance cells used by the robot may be provided by satellite observations (large Cartesian discretisation of the ground) in order to have a rough idea of the terrain nature and by the robot itself to improve its knowledge on some specific areas. Thus, systems maintaining a global environment model and in which different sensors provide different cells discretisation are concerned by this work.

References

[1] R. A. Brooks. Solving the find-path problem by good representation of free space. *IEEE Transactions on Systems, Man, and Cybernetics*, 13(3):190–197, 1983.

[2] R. Chatila. Mobile robot navigation : Space modeling and decisional processes. *ISRR*, 3, 1985.

[3] R. Chatila, M. Devy, S. Lacroix, and M. Herrb. Perception system and functions for autonomous navigation in a natural environment. In *Proceedings of the AIAA/NASA Conference on Intelligent Robots for Field, Factory, Service and Space, Houston, Texas (USA)*, March 1994.

[4] R. Chatila and S. Lacroix. Adaptive navigation for autonomous mobile robot. In *International Symposium on Robotics Research, Munich (Germany)*, October 1995.

[5] M. Hebert. Pixel-based range processing for autonomous driving. In *IEEE International Conference on Robotics and Automation, San Diego, California*, 1994.

[6] K. Kant and S.W. Zucker. Toward efficient trajectory planning: Path velocity decomposition. *International Journal of Robotics Research*, 5:72 – 89, 1986.

[7] T. Lozano-Perez. Spatial planning : A configuration space approach. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(10):681–698, 1983.

[8] J.S.B. Mitchell and C.H. Papadimitriou. The weighted region problem : Finding shortest paths through a weighted planar subdivision. *Journal of the Association for Computing Machinery*, 38(1), 1991.

[9] E. Piat and D. Meizel. Proposal of a probabilistic believes fusion framework – application to range data fusion. In *Proc. of the IROS '97 International Conference on Intelligent Robots and Systems*, pages 1415 – 1422, Grenoble, France, Sept. 1997.

[10] R.F. Richbourg. Solving a two dimensional path planning problem using topographical knowledge of the environment and capability constraints. In *IEEE International Conference on Robotics and Automation, San Francisco (USA)*, 1986.

[11] D. Zhu and J. C. Latombe. New heuristic algorithms for efficient hierarchical path planning. Technical report, Robotics Laboratory, Computer Science Departement, Stanford University USA, 1989. Research Report STAN-CS-89-1279.