Proceedings of International Conference
on Artificial Intelligence

World Modeling by the Fusion of Simpler Models.

Alain Dutech[1] and Manuel Samuelides[1,2]

Email: Alain.Dutech@cert.fr
[1] Centre d'Etude et de Recherche de Toulouse, DERI, 2 av. E. Belin, 31055 Toulouse, France
[2] Ecole Nationale Supérieure de l'Aéronautique et de l'Espace, 10, av. E. Belin, 31055 Toulouse, France.

Abstract:

We simulate a robot moving in a two-dimensional World. Using a range-profile sensor, it builds two simple representations of its environment: a grid-world and a segment-world. A more complex and more precise model of the world is then derived by extracting and fusionning information from these two basic models.

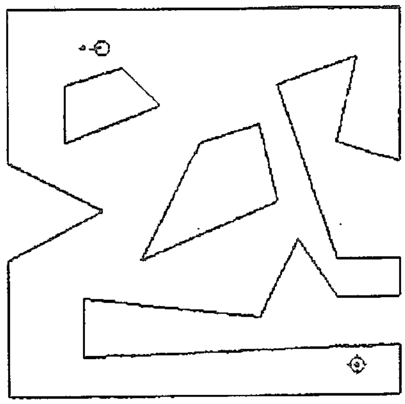
1.0 Introduction.

In mobile robotics, as in many other fields, a good model of the world is always a nice advantage as it is then possible to plan actions and movements. Building and updating such a model is a difficult and tedious task, which is at the center of many research programs ([Le Fur 92], [Moutarlier 92] for example...).

Many technics have been used, as well as many different ways to represent the environment. From grid occupancy of [Moravec 85] to the use of algebraic surfaces [Chatilla 82] and the memorization of interesting routes [Kuipers 79], each representation has some advantages but also its limitations. This is why we decided to fusion the knowledge of two different simple representations of the world to build a precise and usable model of the environment. Similar models are usually built using kalman filtering but, in our case, simpler algorithms lead to very interesting and promising results.

In this paper, we will explain how a simulated robot evolving in a two dimensional world (see Fig. 1), using a range profile sensor described in Sec. 2, first builds two simple representations of its environment. One representation, as described in Sec. 3, is based on a grid to encode the probability of finding an obstacle at a given point in the world. The other model memorizes the environment as a list of segments, as explained in more details in Sec.4. Section 5 deals with the fusion process itself. We finally end by discussing the results and looking towards the future in Sec. 6..

Figure 1: The 2-D simulated world with the robot in the upper-left.



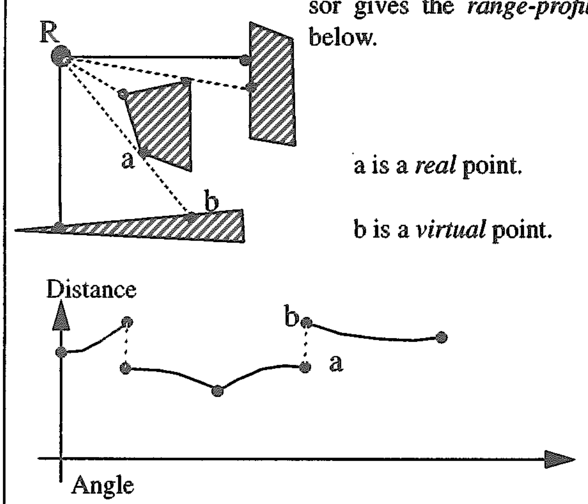
2.0 Data input: the range-profile.

To model the environment, the robot needs some sensory inputs. In our simulation we have chosen to feed it with what we call a *range-profile*. This range profile gives information about the distance to obstacles present in a solid angle in front of the robot. The range profile is a function giving, for each angle belonging to the solid angle, the distance to the nearest obstacle. Figure 2 gives an example of such a profile.

In order for our simulation to be more interesting and useful, we added noise in the perception process. Even if some real captors can treat perceived data so as to obtain a range profile (see [Crowley 85]), it is not easy to model perfectly these sensors, especially if one wants to model the noise. This is why we do not consider our model of the

Figure 2: Range-Profile

The robot in R sees some obstacles and the sensor gives the *range-profile* below.



noise to be realistic, but it nevertheless tries to reproduce some aspects of what could be a real sensor:

- the less perpendicular an obstacle is to the direction of the measure of the distance, the harder it is to get an exact measure.

- it is difficult to get the precise position of an inside corner.

To compensate for the noise coming from the sensors, world modeling modules have used different algorithms. In the next parts, we will see how a grid-world and a segment-world can deal with the noise.

3.0 The Grid-World.

In this first basic model of the world, the robot estimates the probability of an obstacle being present at a given position. In fact we are only estimating this probability value for a finite number of points set on a grid, using a representation similar to [Elfes 86] or [Matthies 88]. After each new perception, the value associated with each point is updated to model the real environment more precisely.

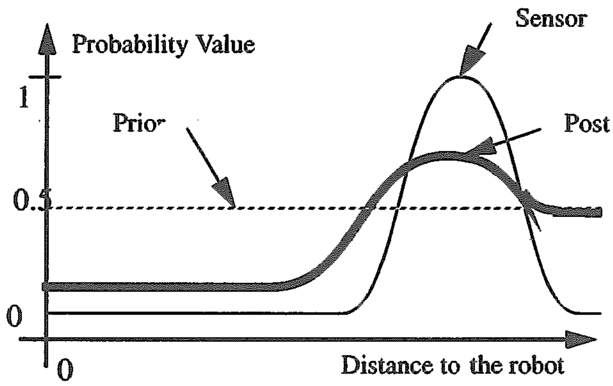
The principle underlying this update process is the Bayes paradigm in which the new estimation of the probability is proportional to the product the probabilities given by the new perception and by the old estimation. This is resumed in the following equation:

$$P_{\text{post}} = \frac{P_{\text{prior}} P_{sensor}}{\text{Normalization}\,Factor}$$

As we know that the sensor is noisy but that the noise model is unknown, we will suppose that the real position of an obstacle is not exactly given by the sensor. In fact the position of the obstacle is given by a random variable that we will assume to be gaussian, with a variance estimated from the sensor parameters and a mean value equal to the measure given by the sensor. Hence, for each perception, we can calculate the probability distribution associated to the observation. Using this distribution and the old estimation of this distribution, the new estimation is computed.

One last detail, as an obstacle prevents the robot from knowing what is behind it, the probability values of points situated too far after the obstacle are set to 0.5, the value for no-knowledge. Figure 3 shows how this would be done for a one-dimensional continuous world.

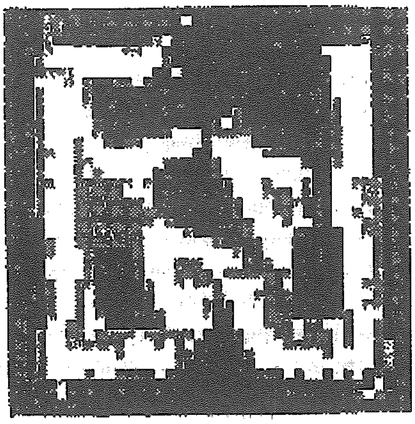
Figure 3: 1-dimensionnal probability distribution encoding the presence of an obstacle.



In our case, the world is a two dimensional grid. But, apart from some minor approximations mainly used to speed up the computation process, nothing else has been added. Figure 4 gives some examples of what we obtained using this grid representation.

As can be seen from Fig. 4, the results are not perfect. Even if we disregard the inherent limitation of using a grid (computation time increases greatly with grid size), the worst drawback lies in the usefulness of the grid. As the abstraction level of the grid is very low, the notions of objects or obstacles are not memorized and it is difficult to deal with dynamic environments, relation and correlations between obstacles and other high level notions.

Figure 4. Grid World: darker points corresponds to a greater probability of abstacle.



4.0 A World of segments.

Starting from the fact that the discontinuities in the range-profile given by the sensor where the easiest points to characterize, we have used another basic representation of the environment for the robot. The obstacles are memorized as a list of segments (connection between two end-points) which are to be correctly positioned despite the noise inherent to the sensors. In fact, noise and uncertainty are the only difficulties in this representation, but they are major problems (see [Hebert 95, 96], [Laumond 85]).

Updating the model after each new perception is the main problem of this kind of representation. It is usually done in a two step algorithm:

- Matching: where each segment of the perception is matched with a segment of the model.

- Updating: using a kalman filter, the position of every obstacle is precised. The correlations between the obstacles are used to enhance this update process.

As we believed that the Fusion would greatly enhance the quality of the model, we decided to only use simple algorithms to update our model. Skipping the matching phase, new segments from the perception are added to the model. Then, segments which are similar (in term of distance and difference of orientation) are melted together. The melting of two or more segment is done very easily by averaging their end-points.

In fact, the averaging process is weighted by the confidence we have in each segment. This confidence represents our certainty about the fact that a segment is related to a segment from the real world and about the precision of its encoding. The confidence value increases when a segment of the model that should be seen by the robot (according to some projections from the robot) is melted with a new segment, meaning it has effectively been perceived by the robot. In the segment was not seen, then its confidence goes down. Segment with a very low confidence are deleted from the model.

Adding to the position of a point, we also memorize if a point is *real* or *virtual*. A virtual point is an end-point of a segment which results of this segment being occulted by another obstacle closer to the robot. The figure 2 gives some examples of real and virtual points. Once again, we will have more confidence in a segment with two real end-points than in any other similar segment.

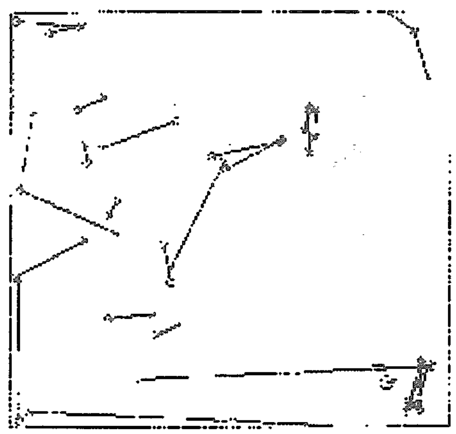
In Fig. 5, two examples of the model of the world are depicted at different stages of the robot trip in its world.

Not unexpectedly, our results are not very good, and this is mostly due to the very simple update algorithms used. As explained and discussed by many ([Durant-Whyte 87], [Voorbraak 95]), the modelization problem is a hard one in the presence of noise. Even if the fact that we are not using any matching procedure is slightly balanced by the frequency at which a new perception is available (the robot has not moved a lot), our decision to use a simple algorithm lead to the model being unable to deal with the noise of the sensors. There is a multiplication of ghost-segments which all represent the same segment from the world and this prevent the system from converging to a stable segment with a high confidence and a precise position.

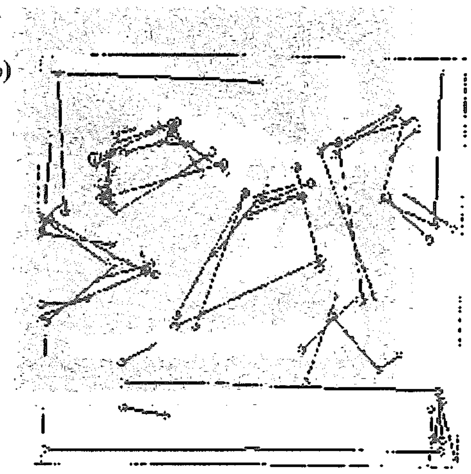
Nevertheless, this model is very interesting as it contains many information on the real world and, most importantly, this knowledge is rather different from the knowledge accumulated by the grid-world. This is quite promising as we want to fusion the two models...

Figure 5: Segment representation of the world, every segment is a real obstacle, in theory. The robot has just made a few moves in (a), a longer travel in (b).

(a)



(b)



5.0 Fusion of the two models.

In order to get a better model of the environment, we want to take all the information we can extract from the two basic models without being restrained by the limitations of these models. This chapter describes the process of Fusion that leads to a global model, the first part focusing on the extraction of useful information from the models and the second part explaining how this information is used.

From the Grid-representation: The grid representation of the world can be seen as a picture of the landscape of the environment, with darker regions being potential obstacles for the robots. We decided to use image processing technics to extract pertinent information from it, a four step treatment:

- Thresholding: the threshold value determines the certainty we will have in the selected regions being obstacles for the robot.

- Filtering and erosion: the position of the obstacles are precised and obstacles that are too small (this is very often due to the noise in the sensor) are discarded.

- Linking of pixels in chains: the obstacles pixels are gathered in chains of pixels, increasing the level of abstraction of the knowledge.

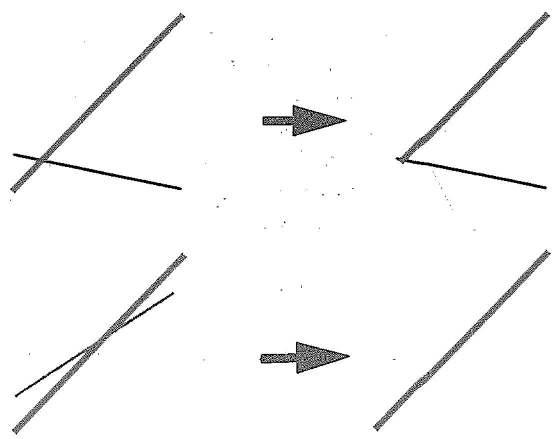
- Polygonal approximation: each chain is approximated by a list of segments so as to be more easily integrated to the knowledge coming from the other model.

Figure 6 describes the process more visually.

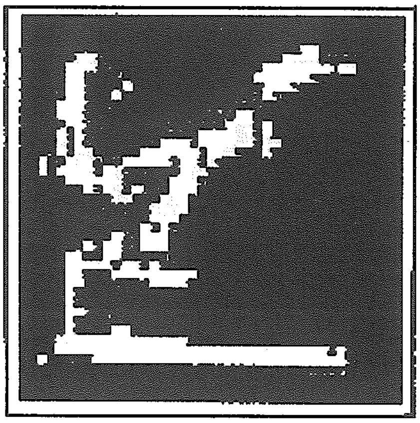
From the segment-representation: The extraction process for the segment-world is easier and straightforward. We already have a list of segment, each with a confidence value. Choosing a confidence level, we first extract all the segments with a higher confidence. If there is a conflict between two segments (usually they intersect each other), they may be altered in order to increase the likelihood of the model.

For example, if both segments intersect near an end-point, the intersection becomes the new end-point (see figure 7). If two segments with similar orientations cross each other, only the most certain is kept.

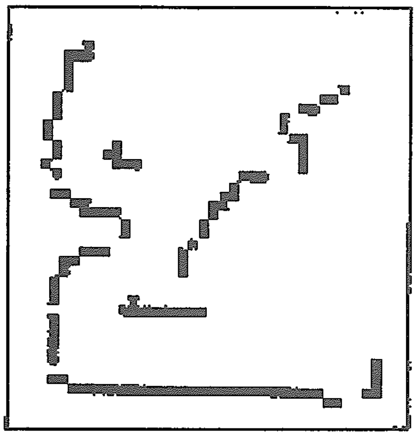
Figure 7: conflicting segments. The larger one has a bigger confidence value...



(a)



(b)



(c)

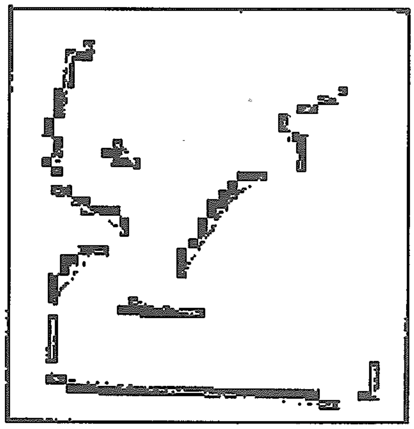


Figure 6: Extraction of segments from the Grid-World.

The original image (a) is thresholded and filtered (b) before a polygonal approximation extraction of obstacles is run on it (c).

This extraction process is illustrated in Fig. 8 where the center obstacle is especially well extracted.

Fusion: The two basic models are now reduced to two lists of segments that are to be fusionned. At this point, some knowledge about he world is incorporated in the system by focusing the Fusion on *corners*. Corners are special end-points where two - and only two - segments meet. As our world is made of polygonal obstacles (see Fig. 1), corners are the basic points to look for.

Corners can be of two kind in our lists of segments:

- every inner point of a list of segment approximating a chain is a corner.

- every end-point which is close to another end-point belonging to a different segment is a corner.

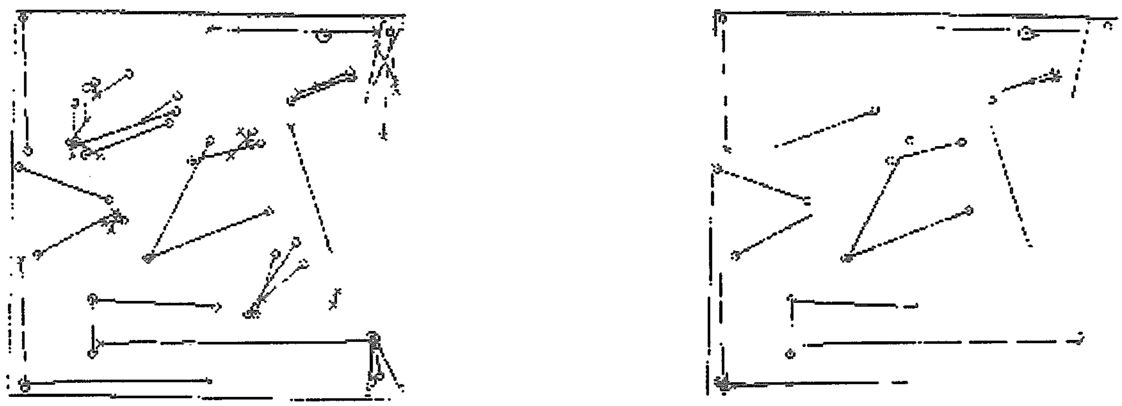
Other end points of the model are then classified in several groups according to their relative distance so as to melt neighboring points in one point. This averaged end-point is either real or virtual, depending on which points it was derived from.

Then, connections between corners and points are evaluated. By now, every corner is linked to several other end-points which can either be corner, real or virtual. The number of such links is reduced to two (as a real corner has only two connections) using a simple set of rules:

- Rule 1: corner-corner connections are better than corner-real or corner-virtual. Of course real-real is better than real-virtual.

- Rule 2: the angle between two connections must not be too close to either 0 or PI (in radiant).

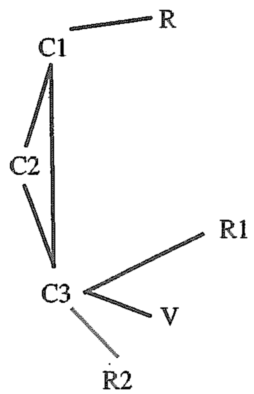
Figure 8: Extraction of information from the segment-world. Before and After...



• Rule 3: a longer connection is better than a smaller one.

Rule 2 and 3 are only used when the application of rule 1 is not enough to get two connections. The figure 9 gives some examples as to how this fusion proceeds.

Figure 9: connections



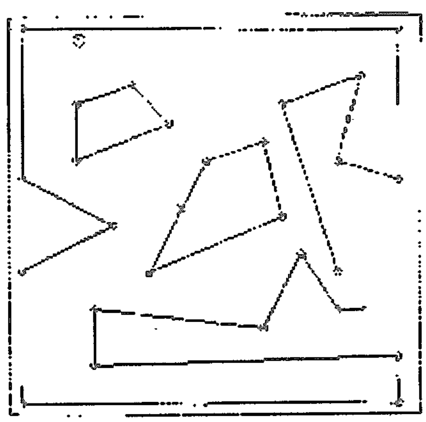
C3 has too much connections. C3-C1 will be preferred over C3-C2 because angles are similar and C3-C1 is longer.

Then C3-R1 is preferred to any other because R1 is more real than V and because the angle between C3-C1 and C3-R1 is preferred over C3-R1 C3-R2.

The resulting model is very satisfying (see Fig. 10) and can be used *as is* by the robot to help it in other tasks (like navigation for example). It is not perfect though, as we sometimes end up with a model with inconsistencies (an obstacle is inside another one, two obstacles intersect each other). These problems are not specifically looked at by now and we only let future perceptions solve the problem by bringing in supplementary information. But, in our future, we plan to implement other functionality to the Fusion process.

Possible Future: The first amelioration would be to incrementally build the global model by integrating the new model obtained by fusion to older ones. Once again we are confronted to a matching and update problem (see Sec. 4) but we believe that these would be easier to solve than in the segment-world model (as demonstrated by our current experiments). Shortly, the facts that each point of the models conveys a lot of information, that the robot does not move a lot between two snapshots of the world and - most importantly - that a newly fusionned model share a lot of information with older models (as it was derived from very similar knowledge sources) lead to a very limited search tree to match corners.

Figure 10: After the fusion, the world is clearer...



Following the human tendency to guess when things are not known, we are trying to incorporate some hypothesis generation and verification modules in our model. When it would seem plausible to have a connection between two end-points, a special hypothesis link would be created and future version of the model would then confirm or infirm this hypothesis. We hope that this should speed up the modelization process, and also increase the capabilities of the robot.

6.0 Conclusions

A simulated robot used different basic representations of the environment to store information about the world received through a range profile sensor. These two representations are then Fusionned in a global world model that takes advantage of the different representation capabilities of each model to bypass their individual limitations. The resulting world model is good enough to help the robot in its navigation task, but could also be used for other kind of tasks. In fact we have incorporated this system in a simulated autonomous agent where the navigation tasks and the modelization task are closely interrelated so as to explore and move in an unknown world.

Future development will aim to increasing the model accuracy by using past knowledge and future prediction on the World. We will also increase the size of the world and use more basic models, either to add other kind of information to the high level model, or to cover different regions of the world with smaller and simpler models that would then be fusionned. We are also planning to use a dynamic world and so that would mean adding new features in the world modeler so as to model moving object at a higher level of abstraction (like a closed chain of segments)

It would also be very interesting, but rather challenging, to use this modelization process on a real robot. This would certainly push us towards using 3D information and this problem is still very open for us...

7.0 Bibliography.

- [Chatilla 82] "Path planning and Environment learning in a mobile robot system", by R. Chatila, in fifth European Conference on Artificial Intelligence, July 1982.
- [Crowley 85] "Navigation for an Intelligent Mobile Robot", by J. Crowley, in IEEE Journal of Robotics and Automation, Vol RA-1, No1, March 1985.
- [Durrant Whyte 87] "Consistent integration and propagation of disparate sensor observations. by H.F Durrant-Whyte in Int Journal of Robotics research, vol 6, No 3, Fall 1987.
- [Elfes 86] "A sonar-based Mapping and Navigation System", by A. Elfes, in IEEE Int Conference on Robotics and Automation, San Francisco, April 1986.
- [Hebert 95] "Probabilistic map learning: Necessity and Difficulties" by P. Hebert, S. Betge-Brezetz and R. Chatila in Int Workshop on Reasoning with uncertainty in Robotics, Amsterdam 95.
- [Hebert 96] "Decoupling geometry and exteroperceptive perception in building global world map of a mobile robot: the use of local maps" by P. Hebert, S. Betge-Brezetz and R. Chatila submitted to IEEE ICRA 96.
- [Kuipers 79] "Common sense knowledge of space: Learning from experience", by B. Kuipers, in Proc of the Sixth International Joint Conference on Artificial Intelligence, 1979.
- [Laumond 85] "A learning system for the understanding of a mobile robot environment" by JP Laumond in Machine and Human Learning.
- [Le Fur 92] "Modelisation geometrique interactive d'environnement pour le pilotage de vehicules d'intervention" by D. Le Fur, These de l'Universite de Rennes I, 1992 (in French).
- [Matthies 88] "Integration of Sonar and Stereo Range Data Using a Grid-Based Representation", by L. Matthies and A. Elfes, in IEEE Int Conference on Robotics and automation, April 1988.
- [Moravec 85] "High Resolution Maps from Wide Angle Sonar", by H. Moravec and A. Elfes, in IEEE International Conference on Robotics and Automation, 1985.
- [Moutarlier 91] "Modelisation autonome de l'environnement par un robot mobile", by P. Moutarlier, in These du LAAS, Toulouse 1991 (in French).
- [Voorbraak 95] "Reasonning with uncertainty in AI" by F. Voorbraak in Int Workshop on Reasoning with uncertainty in Robotics, Amsterdam 95.