HAL
open science

Approche Top-Down de perception multisensorielle : application à la localisation de véhicules

Claude Aynaud, Coralie Bernay-Angeletti, Roland Chapuis, Romuald Aufrère, Christophe Debain

To cite this version:

Claude Aynaud, Coralie Bernay-Angeletti, Roland Chapuis, Romuald Aufrère, Christophe Debain. Approche Top-Down de perception multisensorielle : application à la localisation de véhicules. Reconnaissance de Formes et Intelligence Artificielle (RFIA) 2014, Jun 2014, France. hal-00989213

HAL Id: hal-00989213

https://hal.science/hal-00989213v1

Submitted on 9 May 2014

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.

Approche Top-Down de perception multisensorielle : application à la localisation de véhicules *

C.Aynaud[1] C.Bernay-Angeletti[1] R.Chapuis[1] R.Aufrère[1,2] C.Debain [3]

[1] Institut Pascal, UMR 6602 CNRS/UBP/IFMA
[2] Limos, UMR 6158 CNRS/UBP
[3] Irstea
Campus des Cézeaux, 63171 Aubière Cedex, France

claude.aynaud@univ-bpclermont.fr

Résumé

L'intégration du comportement humain dans le processus d'un robot peut fortement accroître ses capacités, notamment dans le cas d'actions guidées par des objectifs précis et la topologie de l'environnement. L'approche proposée ici présente un système de localisation pour un robot mobile exploitant de manière active à la fois les capteurs dont il dispose et les indices de la carte dans laquelle il doit se localiser. Le robot utilise un réseau bayésien pour la modélisation des différents événements et ainsi atteindre efficacement ses objectifs. Le processus fonctionne en temps réel et a été validé à la fois sur un simulateur et sur des données réelles.

Mots Clef

Localisation, réseau bayésien, processus top-down, intelligence artificielle.

Abstract

Integration of human behavior can greatly enhance the abilities of a robot, specially with a top-down process, guided by the available information. In this paper, a localization system for a mobile robot is proposed, exploiting in the same time the available sensors and the information of map. The robot is using a bayesian network for the modelisation of different events, so it can efficiently achieve its objectives. This process is working in real time and was validated in simulated an real environments.

Keywords

Localization, bayesian network, top-down process, artificial intelligence.

*Ce travail a bénéficié d'une aide de l'État gérée par l'Agence Nationale de la Recherche au titre du programme «Investissements d'avenir», d'une aide de l'Union Européenne (Fonds Européens de Développement Régional –FEDER- de la région Auvergne) et d'une aide de la Région Auvergne dans le cadre du LabEx IMobS3 (ANR-10-LABX-16-01).

1 Introduction

Avec le développement de la robotique, de plus en plus de tâches complexes sont déléguées à des robots, permettant d'effectuer des travaux pénibles, dangereux ou répétitifs pour des êtres humains. L'accomplissement des ces différentes tâches va ainsi amener ces robots à utiliser les ressources à disposition, identifier leur environnement, prendre des décisions, etc. Les robots munis de plusieurs types de capteurs, leur permettant d'agir dans des environnements différents, d'appréhender plus de situations, de résoudre plus de tâches, se développent. Il y a par conséquent un enjeu très important sur la gestion de ces capteurs [7]. Le cerveau humain donne un excellent exemple de management des capteurs, d'exploitation de l'information disponible et d'adaptation à l'environnement, les robots peuvent donc améliorer grandement leurs performances en s'inspirant de l'humain. Plusieurs techniques pour s'en inspirer sont utilisées dans le domaine de l'intelligence artificielle comme les systèmes experts, les réseaux de neurones, les réseaux bayésiens ([14]), etc. Un avantage de ces derniers est qu'ils sont conçus pour travailler en environnement incertain. Ils offrent un modèle permettant d'intégrer différentes informations, de prendre en compte de nouveaux événements [12] et sont donc très pertinents pour le domaine de la robotique [1] et de l'intelligence artificielle [9].

Une application possible de l'intégration d'intelligence artificielle dans un robot concerne les véhicules intelligents. Savoir où se trouve le véhicule est particulièrement important pour des applications comme l'intervention en milieu difficile voire hostile pour l'homme [10], ou encore pour des applications destinées à aider les personnes en difficulté [11] , etc.

Afin d'obtenir une précision de localisation satisfaisante, une méthode souvent utilisée est la perception d'objets de l'environnement, nommés amers [1]. Le principal problème

1. Amer : point de repère fixe et identifiable, dans notre cas cela peut être par exemple des arbres ou des murs

soulevé dans ce type d'approche [16] est celui de l'association de données, c'est-à-dire associer une détection perçue avec l'objet réel. En contrepartie, ce type d'approche peut être efficace et nécessite peu de calculs. Plusieurs façons de remédier à ce problème d'association existent. Dans [5], des amers actifs émetteurs sont utilisés, chacun d'entre eux possédant un moyen d'identification unique. Le principal inconvénient des amers actifs est la nécessité d'équiper l'environnement. L'utilisation exclusive d'amers passifs est moins contraignante. Dans ce cas, les mauvaises associations de données peuvent être nombreuses et nécessitent un traitement adapté. Certaines méthodes populaires comme le filtre à particules [13] [6] propagent à chaque instant plusieurs hypothèses d'association de données et donc de localisation. L'inconvénient de ce type de méthodes est le coût calculatoire ; il serait judicieux de développer une méthode permettant de remettre en cause facilement une mauvaise association.
Pour la localisation, deux grandeurs sont importantes : la précision et l'intégrité [2]. Les deux ne peuvent être garanties pour un coût raisonnable avec uniquement des capteurs de type GPS. Il est donc nécessaire de s'orienter vers d'autres voies visant à combiner plusieurs sources d'informations.
Cet article, dans la continuité de [2], propose une méthode basée sur les réseaux bayésiens pour la localisation de véhicules en environnement naturel. Les capteurs, les détecteurs et les amers sont choisis en fonction de leur pertinence et de leur apport en matière de précision et/ou d'intégrité, ce pour éviter de fusionner a posteriori des données peu pertinentes. Suivant l'environnement et les capteurs disponibles, le système agira différemment en s'adaptant au mieux à chaque situation.

2 Approche top-down proposée

Le problème posé est général et peut se formuler ici de la façon suivante : "Comment gérer au mieux toutes les ressources dont on dispose pour atteindre l'objectif ?". Le processus mis en place ici est un processus top-down similaire à celui mis en place dans [3] ou [4] dans un tout autre contexte (reconnaissance d'objets dans une image).
Le processus global est représenté par le schéma de la figure 1.Dans notre application, l'objectif est alors précisé de la manière suivante : le véhicule doit connaître sa position avec une précision donnée (par exemple plus ou moins 10cm) et une certaine confiance (par exemple une confiance de 90%). Ici, le calcul a été modifié par rapport à [3] ou [4], notamment par l'introduction d'un réseau bayésien permettant une plus grande souplesse et une meilleure appréhension de la méthode. Pour atteindre cet objectif, le système dispose notamment d'une carte de l'environnement répertoriant différents amers géolocalisés (arbres, murs, routes, etc.) et leurs caractéristiques. La problématique est alors la suivante : dans une configuration donnée, quel amer le véhicule doit-il essayer de détecter afin de répondre au mieux aux objectifs ? Avec quel capteur ? Avec quel détecteur ? Une fois la détection demandée, comment interpréter le résultat retourné par le détecteur ?
A un instant t donné, une certaine estimation de la position du véhicule est disponible. Pour cette position, le processus doit déterminer s'il est utile de percevoir l'environnement et si oui quel est le meilleur triplet perceptif (un amer, un détecteur, et un capteur, voir 3) disponible. Pour bien comprendre l'utilité du triplet, considérons un humain qui désire se déplacer d'une pièce à une autre va généralement choisir ses yeux comme capteur, mais pour peu qu'il y ait une faible luminosité ce capteur devient totalement inopérant, la personne va se déplacer à tâtons et a donc changé de capteur en sélectionnant le toucher. Notre méthode agit exactement de la même façon, les capteurs et les détecteurs ne remonte pas d'informations en continu comme le ferait une approche exclusivement bottom-up mais seulement si en amont ils ont été jugés utiles dans cette situation.
Une fois le triplet choisi, le système sollicite alors le capteur concerné et lance le processus de détection. Dès que ce dernier est réalisé, une mise à jour de la position et de la confiance [3] associée est réalisée.
Dans le cadre de cet article, nous nous focalisons sur le choix du triplet perceptif et sur la modélisation bayésienne qui constituent le cœur de notre système.

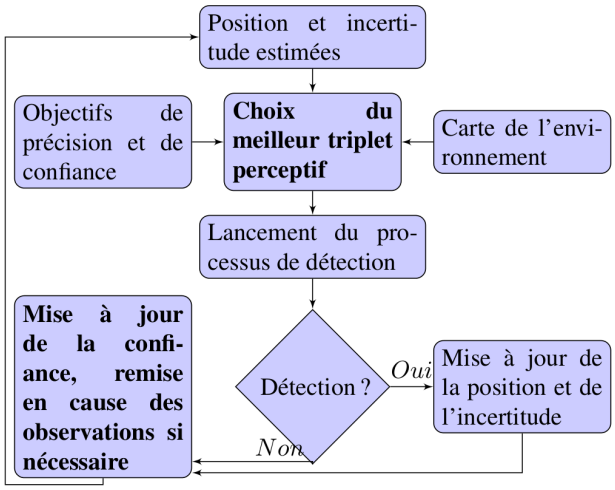


FIGURE 1 – Processus d'observation de l'environnement à un instant donné

3 Le triplet perceptif

3.1 Définition et utilité du triplet

La notion de triplet perceptif utilisée ici est telle que définie dans [15]. Un triplet perceptif est constitué d'un amer, d'un détecteur et d'un capteur. L'amer est typique de notre environnement, le capteur est typique de notre véhicule (positionnement sur le véhicule, portée, résolution, ...), et le dé-

2. Intégrité : Le système est intègre quand la position réelle du véhicule appartient à l'espace d'incertitude autour de la position estimée

3. Confiance : Probabilité que la position vraie du véhicule soit comprise dans l'intervalle donné par l'incertitude de la position estimée, soit la probabilité d'être intègre

tecteur est typique de notre application. Le triplet perceptif intègre donc beaucoup plus d'informations qu'un amer seul. Pour pouvoir sélectionner avec soin le meilleur triplet perceptif, un objectif à atteindre en terme de précision (par exemple 10 cm) et de confiance sur notre estimation est défini. Certains triplets sont peu intéressants parce qu'ils n'améliorent pas notre estimation ou parce que l'amer est difficilement observable avec ce capteur. Pour toutes ces raisons, une première étape de sélection est réalisée. Sur la figure 2, une illustration de ce principe est présentée. Un véhicule est positionné au point O avec une incertitude de position représentée par l'ellipse rouge et différents amers (points A, B, C, D et E) sont également disposés. Le

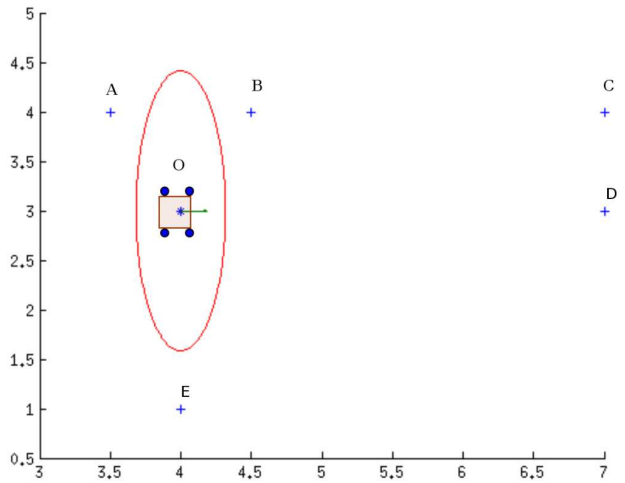


FIGURE 2 – Exemple de scénario

véhicule est orienté vers la droite et possède ici un capteur lui permettant d'avoir un angle de vue de -90 à 90 degrés (il "voit" donc tout ce qui se trouve devant lui). Pour l'exemple, nous considérons que le véhicule ne possède qu'un seul capteur et qu'un seul détecteur, les triplets perceptifs peuvent donc se réduire aux amers puisque le reste est à l'identique. Chacun de ces amers possède des avantages et inconvénients vis-à-vis de l'objectif. Le système est ici capable de détecter la distance d du véhicule aux amers. Quand un amer est détecté, il est alors possible de dire qu'il se trouve dans la zone d'intersection d'un cercle de rayon d centré sur l'amer et de la zone d'incertitude de la position estimée. Le point A n'est jamais visible, il n'est donc pas pertinent de le choisir. Les points B, C ou D sont ceux qui sont le plus facilement observables mais, de par l'orientation de l'ellipse d'incertitude les points C et D sont ceux qui améliorent le moins la précision. Le point E est celui qui améliorerait le mieux la précision du véhicule, mais il a très peu de chances d'être observable. Finalement, l'amer B est celui qui offre le meilleur compromis et qui doit être choisi par le système.

4 Critère de sélection des triplets

Afin de sélectionner le triplet le plus pertinent, c'est-à-dire qui nous offrira le meilleur compromis vis-à-vis de nos objectifs, à savoir précision et intégrité, un critère de sélection est défini pour chaque triplet possible. Le triplet qui est associé au meilleur critère est alors choisi.

4.1 Définition du critère de sélection

Le critère utilisé ici mixe la précision Pp apportée par le triplet, la probabilité de détecter effectivement l'amer recherché $P(db)$ et la probabilité d'avoir une nouvelle information (caractérisé par $P(\overline{CI_k})$) :

$$C = Pp \times P(db) \times (P(\overline{CI_k})) \tag{1}$$

Ce critère possède les propriétés suivantes :
- Un triplet n'apportant aucune information nouvelle n'est pas sélectionné ;
- Un triplet qui a très peu de chances d'être détecté verra son critère faible ;
- A probabilité de bonne détection et apport d'information égaux le triplet améliorant le mieux la précision est sélectionné.

4.2 Calcul du critère

Le détail du calcul de Pp est donné dans [2]. Le calcul de $P(\overline{CIk})$ est effectué en fonction de la distance parcourue depuis la dernière détection de ce triplet. Plus la distance est importante, plus l'information est nouvelle. Enfin le calcul de $P(db)$ sera effectué à l'aide du réseau bayésien présenté dans la section suivante.

5 Modélisation bayésienne

Les réseaux bayésiens permettent de formaliser la connaissance dans un environnement incertain et sont donc tout à fait indiqués dans ce cas. Un réseau bayésien nous permettra de calculer la probabilité $P(db)$ de réaliser une bonne détection puis de calculer la confiance $P(O_k)$ à avoir dans notre estimation de la position.

5.1 Événements pris en compte

Les événements modélisés dans le réseau bayésien sont :
- O_k^- : la localisation avant la mise à jour suite à la détection de l'amer est intègre ;
- z : l'amer est observable avec le détecteur et le capteur sélectionnés ;
- A_z : au moins un amer est présent dans la zone observée ;
- d_k : un amer a été détecté
- db : un amer a été détecté et l'association avec la carte est correcte ;
- CI_k : le système a déjà essayé de détecter cet amer précédemment ;
- dp : le système a déjà essayé de détecter cet amer précédemment et avait détecté quelque chose ;
- O_k^+ : la localisation est intègre après la mise à jour ;

Avec tous ces événements, il est possible de construire un réseau bayésien qui prenne en compte en compte les performances des détecteurs, l'occultation des amers, l'ambiguïté entre amers, la taille de la zone de recherche.

5.2 Architecture du réseau

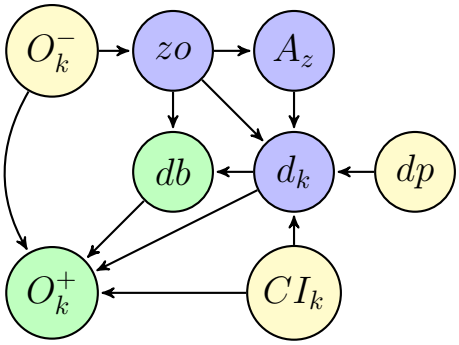


FIGURE 3 – Réseau bayésien . Les nœuds jaunes sont les entrées (leur probabilité est connue). Les nœuds verts sont les événements dont nous voulons calculer les probabilités.

Ce réseau bayésien nous permet de modéliser les liens entre les différents éléments. Il est décrit dans la section suivante mais nous pouvons déjà noter que la structure est strictement la même pour chaque triplet, quelque soit le type de capteur, de détecteur ou d'amer.

5.3 Tables de probabilités

Les probabilités sont calculées pour chaque triplet en prenant en compte ses caractéristiques propres. Pour toutes les tables, ϕ peut prendre la valeur 0 ou 1.

zo.

O_k^-	zo
0	0
1	obs

La première ligne signifie que si la position estimée du véhicule est fausse, l'amer recherché ne sera pas visible. Dans la seconde ligne, obs est la probabilité que l'amer puisse être vu si la position estimée du vahicule est vraie. Dans l'exemple 2, pour les amers B, C et D $obs = 1$ parce qu'ils sont toujours détectables où que soit le véhicule dans l'ellipse rouge. A l'inverse, $obs = 0$ pour A parce qu'il est derrière le véhicule et que le capteur ne voit que vers l'avant. Pour E, $obs = 0.5$ parce que l'amer est visible seulement si le véhicule est dans la moitié droite de l'ellipse rouge.

A_z.

zo	A_z
0	γ
1	1

La seconde ligne affirme juste que si l'amer est observable il y a au moins un amer dans la zone observée. La première ligne montre que si l'estimation de la position est mauvaise, il est malgré tout possible qu'il y ait d'autres amers de même type dans la zone observée. Par exemple, si le véhicule est dans une forêt et veut détecter un arbre dans une grande zone, $\gamma \approx 1$. Si le véhicule est dans un désert, $\gamma \approx 0$. γ est calculée à partir de la densité d'amers dans la carte et la taille de la zone observée.

d_k.

CI_k	dp	z	A_z	d
0	ϕ	0	0	β
0	ϕ	0	1	$1 - \alpha^{AverageNbLandmark}$
0	ϕ	1	ϕ	$1 - \alpha^{AverageNbLandmark}$
1	0	ϕ	ϕ	0
1	1	ϕ	ϕ	1

Les deux dernières lignes représentent le fait que si l'information est totalement corrélée, cela signifie que le détecteur renverra exactement le même résultat (notons que l'information ne peut pas être corrélée si le triplet n'a pas déjà été utilisé avant). Sur la première ligne, β est la probabilité que le détecteur renvoie un résultat alors qu'il n'y a rien. Dans les deux lignes suivantes, α est la probabilité que le détecteur ne détecte rien alors qu'il y a des amers compatibles dans la zone. Ainsi chaque amer présent (et il y en a environ $AverageNbLandmark$) dans la zone a une probabilité α de ne pas être vu, par conséquent la probabilité que le détecteur ne détecte aucun amer est de $\alpha^{AverageNbLandmark}$. Par suite, la probabilité que le détecteur trouve quelque chose est de $1 - \alpha^{AverageNbLandmark}$.

d_b.

zo	d	db
0	ϕ	0
1	0	0
1	1	$\frac{1}{nbNeighbor}$

La première ligne représente le fait que si l'amer n'est pas observable, même si le détecteur a renvoyé un résultat l'amer recherché n'a pas été détecté. La deuxième ligne est évidente : si la détection a échoué, il est impossible que le détecteur ait détecté le bon amer. La troisième ligne représente le fait que l'association entre l'amer recherché et le résultat du détecteur est mauvaise : le détecteur a peut être détecté un voisin de l'amer recherché à sa place.

O_k^+.

O_k-	d	db	O_k^+
0	ϕ	ϕ	0
1	0	ϕ	1
1	1	0	0
1	1	1	1

La première ligne signifie que si la position estimée n'était pas bonne, elle ne le deviendra pas avec la mise à jour. La deuxième ligne représente le fait que si le détecteur n'a rien trouvé, la position estimée ne sera pas mise à jour donc si elle était bonne elle le restera. La dernière ligne dit que si la détection réussit, la position estimée après la mise à jour sera bonne seulement si nous avons fait la bonne association, c'est-à-dire seulement si db est vrai.

6 Mise à jour

6.1 Détection de l'amer

Une fois le triplet le plus pertinent sélectionné, la phase de détection débute. Étant donnée l'incertitude sur la position estimée du véhicule et la position de l'amer à rechercher (position donnée par la carte), le véhicule sait dans quelle zone regarder. Il va donc pouvoir se focaliser sur cette zone. Ceci est illustré sur la figure 4 où il est possible de voir d'un

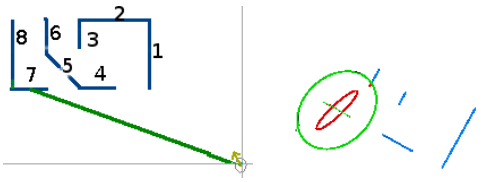


FIGURE 4 – Focalisation sur le mur 7

côté une carte avec la position du robot, son ellipse d'incertitude, les amers présents (en l'occurrence huit murs), un trait vert reliant la position estimée du véhicule et le mur sélectionné et d'un autre côté les données provenant du capteur. Dans l'espace du capteur (un télémètre laser ici), l'ellipse rouge représente la zone où le robot situe le centre du mur recherché. L'ellipse verte, calculée à partir de l'ellipse rouge et des caractéristiques du mur (longueur et orientation), représente la zone dans laquelle le détecteur va rechercher le mur. Ces données sont disponibles en utilisant de façon optimale l'ensemble des informations mises à notre disposition. Cela permet d'augmenter les performances du détecteur en améliorant le rapport signal sur bruit. Remarquons que ces performances augmentent au fur et à mesure des détections réalisées puisque la position du robot se précise et donc que la zone de focalisation devient de plus en plus petite.

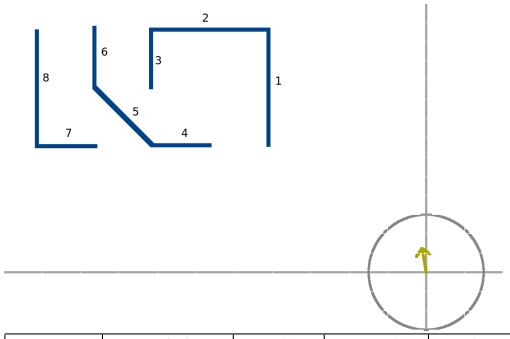
6.2 Mise à jour de la confiance

A chaque essai de détection, la probabilité d'être réellement dans la zone d'incertitude de la position estimée (c'est-à-dire d'être intègre) est tenue à jour. Cette mise à jour est effectuée grâce au réseau bayésien. En effet, à cette étape l'événement d_k est connu, cela nous permet d'intégrer le résultat correspondant dans le réseau (ce qui signifie mettre $P(d_k)$ à 1 si quelque chose a été détecté, 0 sinon).

7 Résultats

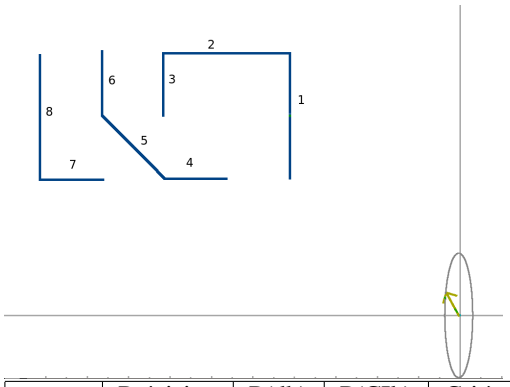
7.1 Évolution du critère dans un cas statique

Nous allons voir ici sur un cas simple (figures 5 et 6) l'évolution du critère et donc la sélection du triplet qui en ressort. Un véhicule est positionné au point de coordonnées $(0, 0)$ et l'incertitude de sa position est représentée par une ellipse. L'environnement est constitué d'amers de même type à savoir des murs, numérotés ici de 1 à 8. Au départ (figure 5), l'ellipse d'incertitude est un cercle, le détecteur est le même pour tous, l'objectif à atteindre ne privilégie aucune direction, chaque triplet a donc le même apport en précision. Aucun amer n'a encore été détecté, aussi ils apportent tous une information totalement nouvelle. Le triplet qui va être sélectionné sera donc celui qui a la meilleure probabilité d'être détecté, cette probabilité est calculée en prenant en compte l'occultation par un autre amer (le mur 2 est occulté par le mur 1), la confusion possible avec un autre amer (le mur 4 peut être confondu avec le mur 7). Ainsi c'est le mur 1 qui est sélectionné.
A l'étape 2 (figure 6), l'incertitude n'est plus isotrope, aussi tous les triplets ne vont pas améliorer la position de la



	Précision	P(db)	P(CIk)	Critère
Mur 1	0.24	1	0	0.24
Mur 2	0.24	0	0	0
Mur 3	0.24	0.23	0	0.055
Mur 4	0.24	0.5	0	0.12
Mur 5	0.24	0.09	0	0.02
Mur 6	0.24	0	0	0
Mur 7	0.24	0.37	0	0.088
Mur 8	0.24	0	0	0

FIGURE 5 – Évolution du critère : Étape 1



	Précision	P(db)	P(CIk)	Critère
Mur 1	0.27	0.99	0.99	0.0026
Mur 2	0.67	0	0	0
Mur 3	0.27	0.59	0	0.159
Mur 4	0.67	0.49	0	0.328
Mur 5	0.56	0	0	0
Mur 6	0.27	0	0	0
Mur 7	0.67	0.48	0	0.321
Mur 8	0.27	0	0	0

FIGURE 6 – Évolution du critère : Étape 2

même façon. L'amer 1 a déjà été détecté, il ne nous apporte quasiment plus aucune information. En revanche, les probabilités de bonne détection des autres amers ont été précisées. Finalement, c'est l'amer 4 qui est sélectionné.

7.2 En environnement simulé

Pour appliquer ces principes, un simulateur indépendant a été utilisé. Dans ce simulateur, il est possible de créer notre propre environnement et de simuler un véhicule équipé de

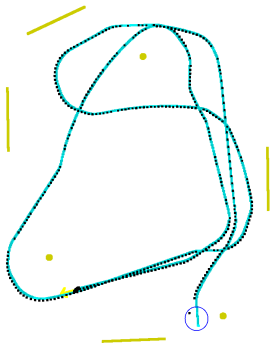


FIGURE 7 – Résultats en environnement simulé (le point de départ est dans le cercle, en noir la trajectoire estimée, en cyan la trajectoire réelle, en jaune les amers (colonnes ou murs))

différents capteurs. Il est ainsi possible de comparer la trajectoire estimée avec la trajectoire réelle. Le véhicule est ici équipé d'un télémètre laser, et de capteurs odométriques [4]. Les amers utilisés sont des colonnes ou des murs. Des détecteurs de segments et de cercles sont fournis pour détecter les amers. La trajectoire estimée (figure 7) a un écart

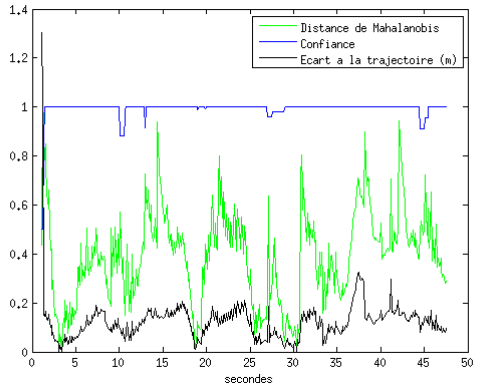


FIGURE 8 – Performances de la localisation

moyen de 12 cm à la trajectoire réelle (figure 8). Sur la figure 8, nous pouvons voir la confiance estimée par l'algorithme au cours du temps, qui estime que la probabilité que la position soit intègre est supérieure à 70% et même égale à 100% durant très longtemps. En vert, est représentée la distance de Mahalanobis entre la position estimée et la position réelle, si cette distance est plus petite que 1 alors nous pouvons considérer que la position est intègre, c'est effectivement presque toujours le cas. La confiance a donc exactement le comportement souhaité. Les résultats sont satisfaisants.

7.3 En environnement réel

Les résultats en environnement simulé ont permis de valider le concept dans un environnement favorable où tout est contrôlé. Le passage en environnement réel permet de voir si notre modélisation bayésienne est suffisamment complète pour avoir de premiers résultats.

4. Mesurent les déplacements des roues

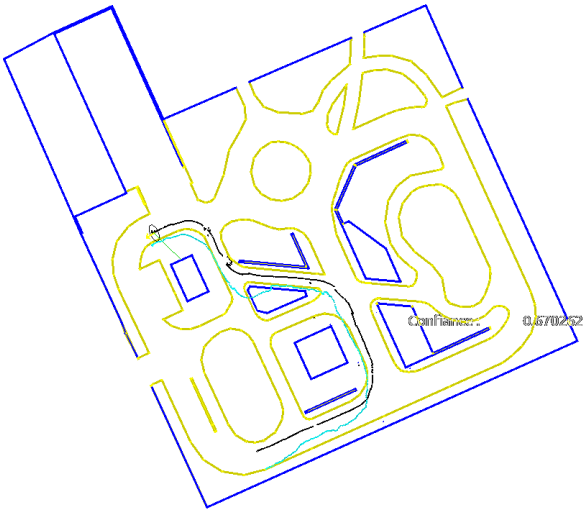


FIGURE 9 – Résultats en environnement réel (en noir la trajectoire estimée, en cyan la trajectoire donnée par le GPS, en jaune les trottoirs et en bleu amers détectables (poteaux , murs))

Sur les résultats réels (figure 9), nous pouvons voir la différence entre un GPS naturel bas coût avec une précision de l'ordre du mètre et notre estimation. Le GPS donne une trajectoire du véhicule très imparfaite (avec notamment des passages sur les trottoirs, dans les bâtiments, etc). La trajectoire fournie par notre système de localisation est beaucoup plus représentative de la réalité.

8 Conclusion

Les résultats de cette méthode sont très prometteurs et valident la pertinence d'une approche top-down pour l'application de la localisation de véhicules. Le système réagit de manière efficace en fonction de ses objectifs et des données disponibles. Le réseau bayésien de par sa structure permet une grande flexibilité sur les paramètres et le rajout facile d'un nouvel événement comme une déconnexion de capteur par exemple. Le principe générique de la méthode permettra également dans cette application de rajouter de nouveaux capteurs comme la caméra par exemple. L'environnement pourra être pris en compte de manière plus fine, permettant d'intégrer d'autres événements tels la présence d'amers non géoréférencés, un capteur défectueux, un amer répertorié mais non détectable...

Références

[1] Nisar Ahmed et al. Towards cooperative bayesian human-robot perception : Theory, experiments, opportunities. In *Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence*, Washington, 2013.

[2] Claude Aynaud, Coralie Bernay-Angeletti, Romuald Aufrère, Christophe Debain, and Roland Chapuis. Sélection des données et regard critique sur le résul-

tat dans le cadre de la localisation de véhicules. In *GRETSI'2013*, Brest, France, September 2013.

[3] Coralie Bernay-Angeletti, Claude Aynaud, Romuald Aufrère, Roland Chapuis, et al. Stratégie de perception active pour l'interprétation de scènes : Application à une scène routière. In *Orasis, Congrès des jeunes chercheurs en vision par ordinateur*, Cluny, 2013.

[4] Roland Chapuis, Frederic Chausse, and Noel Trujillo. Progressive focusing : A top down attentional vision system. In *Advances in Visual Computing*, pages 468–477. Springer, 2008.

[5] Leonimer Flávio de Melo, João Mauricio Rosário, and Almiro Franco da Silveira Junior. Mobile robot indoor autonomous navigation with position estimation using rf signal triangulation. *Positioning*, 4 :20–35, 2013.

[6] A Dean, R Martini, and S Brennan. Terrain-based road vehicle localisation using particle filters. *Vehicle System Dynamics*, 49(8) :1209–1223, 2011.

[7] Alfred O Hero and Douglas Cochran. Sensor management : Past, present, and future. *Sensors Journal, IEEE*, 11(12) :3064–3075, 2011.

[8] Davis E. King. Dlib-ml : A machine learning toolkit. *Journal of Machine Learning Research*, 10 :1755–1758, 2009.

[9] Kevin B Korb and Ann E Nicholson. *Bayesian artificial intelligence*. CRC press, 2003.

[10] Geert-Jan M Kruijff et al. Designing intelligent robots for human-robot teaming in urban search & rescue. In *Proceedings of the AAAI Spring Symposium on Designing Intelligent Robots*, Stanford University, 2012.

[11] Ruijiao Li et al. Ros based multi-sensor navigation of intelligent wheelchair. In *Fourth International IEEE Conference on Emerging Security Technologies (EST),*, pages 83–88, 2013.

[12] Kevin Patrick MURPHY. *Dynamic Bayesian Networks : Representation , Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.

[13] Ioannis M Rekleitis. A particle filter tutorial for mobile robot localization. *Centre for Intelligent Machines, McGill University, Tech. Rep. TR-CIM-04-02*, 2004.

[14] Stuart Jonathan Russell, Peter Norvig, John F Canny, Jitendra M Malik, and Douglas D Edwards. *Artificial intelligence : a modern approach*, volume 74. Prentice hall Englewood Cliffs, 1995.

[15] Cédric Tessier, Christophe Debain, Roland Chapuis, and Frédéric Chausse. Map aided localization and vehicle guidance using an active landmark search. *Information Fusion*, Volume 11, Issue 3 :283–296, 2010.

[16] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.