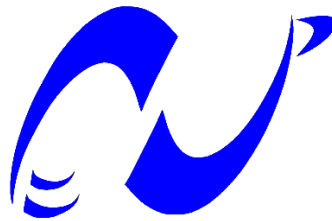


REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE

Département d'Automatique



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Mémoire de Master en Automatique

THEME :

**Cartographie et localisation simultanée pour véhicules
autonomes**

Réalisé et présenté par :

OUALA Saïd

Dirigé par :

M. H. ACHOUR

Juin 2015

École Nationale Polytechnique, 10 Avenue Hassen Badi, El-Harrach, Alger. Algérie

Remerciements

En premier lieu, Je remercie Dieu tout puissant de m'avoir donné le courage, la volonté et la patience pour réaliser ce travail.

Je remercie mes parents, qui m'ont soutenu tout au long de mes études.

Ce modeste travail, fruit de mon cursus, n'a été possible que grâce au concours de tous mes enseignants dont je loue les efforts qu'ils ont consentis durant toutes ces années.

Dédicace

Je dédie ce modeste travail en premier lieu à mes très chers parents qui m'ont toujours soutenu, aidé et encouragé. Qu'ils trouvent ici le témoignage de ma reconnaissance la plus dévouée.

À mon frère et mes sœurs ;

À tous les membres de ma grande famille ;

À tous mes ami(e)s et camarades.

À tous ceux qui m'ont aidé de près ou de loin pour réaliser ce modeste travail.

Saïd

Résumé

Un robot mobile autonome doit percevoir correctement l'environnement où il évolue. Autrement, l'interprétation qu'il en fait sera erronée et ne peut prendre des décisions correctes. La perception est également un point clé pour tout véhicule intelligent ou fournissant des fonctions d'assistance à la conduite. Dans ce cadre, la perception peut être divisée en deux parties : la première appelée SLAM (*Simultaneous Localization And Mapping*) s'intéresse à la construction d'une carte de l'environnement et à la localisation du véhicule dans cette carte, et la deuxième partie, appelée DATMO (*Detection And Tracking on Moving Objects*), traite la détection et du suivi des objets mobiles dans l'environnement.

L'objectif de ce travail est de simuler plusieurs méthodes de SLAM dans le but de conclure sur la solution la plus efficace permettant d'assurer une cartographie et localisation simultanée acceptable quel que soit l'environnement tout en assurant une perception artificiel au point.

Mots clés : Robot mobile, cartographie et localisation simultanés, SLAM.

Abstract

An autonomous mobile robot must correctly perceive the environment in which it operates. Otherwise, the interpretation will be wrong and the robot cannot make correct decisions. Perception is also a key point for any intelligent vehicle or even for a vehicle providing only driver assistance functions. In this context, perception can be divided into two stages: the first part called SLAM (*Simultaneous Localization And Mapping*) is concerned with building an online map of the external environment and localizing the host vehicle in this map, and second part, called DATMO (*Detection And Tracking of Moving Objects*), deals with finding moving objects in the environment and tracking them over time.

The objective of this work is to simulate some SLAM methods in order to conclude on the most effective solution to ensure acceptable simultaneous localization and mapping regardless of the type of the environment while providing efficient artificial perception.

Keywords : Mobile robot, Simultaneous Localization And Mapping, SLAM.

ملخص:

الروبوت المحمول المستقل يجب ان يدرك المحيط الذي يعمل فيه بشكل صحيح و الا سيكون التفسير المقدم منه خاطئا ولن يتمكن من اتخاذ قرارات صحيحة. كما يعتبر الإدراك مفتاح لأية مركبة ذكية أو لتلك التي تقدم مهام المساعدة للقيادة. و في هذا السياق، يمكن تقسيم الإدراك إلى قسمين: SLAM و DATMO.

إن الهدف من هذا العمل هو محاكاة بعض أساليب SLAM من أجل الوصول إلى الحل الأنجع الذي يضمن رسم الخرائط و تحديد متزامن مقبول، و هذا بغض النظر عن المحيط، مع ضمان إدراك اصطناعي فعال.

الكلمات الدالة: الروبوت المحمول، رسم الخرائط و التحديد المتزامن، SLAM.

Table des matières

Introduction générale.....	1
Chapitre 1	3
1.1 Robot mobile autonome.....	3
1.2 Applications de la robotique mobile.....	3
1.3 Navigation autonome des robots mobiles.....	4
1.3.1 Perception artificielle des véhicules autonomes	5
1.3.2 Présentation et définition du SLAM.....	8
1.4 Conclusion.....	8
Chapitre 2	9
2.1 Méthode de Modélisation de l'environnement pour le SLAM.....	9
2.1.1 Méthodes de modélisation métriques de l'environnement.....	9
2.1.2 Méthodes de modélisation topologique	11
2.2 Les différentes approches de cartographie et localisation simultanée (SLAM).....	12
2.2.1 Approches probabilistes.....	12
2.2.2 Scan-matching SLAM.....	20
2.3 Conclusion.....	22
Chapitre 3	23
3.1 Simulation sous ROS	23
3.2 Simulations sous MATLAB	25
3.2.1 Simulation de la méthode EKF-SLAM.....	25
3.2.2 Simulation de la méthode UKF-SLAM.....	31
3.2.3 Simulation de la méthode ICP-SLAM	37
3.3 Conclusion.....	40
Conclusion générale.....	42
Bibliographie	44

Table des figures

1. Figure 1-1: Les deux composantes de la perception : SLAM et DATMO.	5
2. Figure 1-2 : L'objet de la perception en robotique. A droite : l'environnement réel. A gauche : le résultat de l'inférence des données capteur par le robot.....	5
3. Figure 1-3 : La perception utilise les données des capteurs en entrée et produit en sortie : l'état du véhicule (position et vitesse, ...), la carte de l'environnement et une liste des objets mobiles et leurs caractéristiques cinématiques.	7
4. Figure 1-4 : Architecture PDA d'un système robotique.....	7
5. Figure 2-1: Carte géométrique de l'environnement.....	10
6. Figure 2-2: Représentation de l'environnement avec une grille d'occupation.....	11
7. Figure 2-3: comparaison entre les deux catégories de représentation de l'environnement, A gauche : représentation topologique de l'environnement, à droite : représentation métrique.....	12
8. Figure 2-4: Représentation du problème de SLAM : une estimation simultanée de la position du robot et de celle des points de l'environnement.....	13
9. Figure 2-5 : Environnement réel.....	15
10. Figure 2-6 : EKF SLAM.....	16
11. Figure 2-7 : UKF SLAM.....	16
12. Figure 2-8 : (a) trajectoire du véhicule prédite par odométrie, (b) Trajectoire réelle (en pointillée) et trajectoire à partir de l'algorithme Fast-SLAM (en rouge), (c) résultats de l'application du Fast-SLAM en environnement externe.....	17
13. Figure 2-9 : robustesse du Fast-SLAM et de l'EKF-SLAM aux erreurs de l'odométrie.....	18
14. Figure 2-10 : SLAM en utilisant une seule carte.....	19
15. Figure 2-11 : DP-SLAM.....	19
16. Figure 2-12 : Résultat de la méthode F2F-SLAM.....	20
17. Figure 2-13 : Résultat de la méthode P2F-SLAM.....	21
18. Figure 2-14 : Deux scans avant et après la mise en correspondance en utilisant la méthode ICP-SLAM.....	22
19. Figure 3-1 : Résultat de l'application de l'algorithme ICP.....	37

Introduction générale

La conception des robots mobiles autonomes est un domaine de recherche en pleine expansion. En effet, d'après *l'International Federation of Robotics*, le nombre de robots mobiles recensés dans le monde a dépassé le million d'unités fin 2010. Ces véhicules sont largement utilisés dans l'industrie, comme moyen de transport et même dans l'exploration planétaire.

Actuellement, les milieux occupés par les robots ont fortement tendance à s'extrapoler à des environnements de bureaux ou à des environnements domestiques. Les types d'applications possibles deviennent alors innombrables. Cela peut être des tâches de nettoyage et d'entretien, une assistance à une personne handicapée dans des tâches d'exploration et de préhension, le guidage lors d'une visite de musée, etc. On parle alors, de façon générale, de « robotique d'intérieur ».

Un tel cadre d'utilisation requiert un niveau minimum d'autonomie et de facilités de navigation pour le système robotisé. Pour ce faire, le robot est doté de capacités de perception et d'information sur son environnement qui lui permettent de se mouvoir en autonomie, sans se perdre, tout en évitant les obstacles. La perception apparaît ainsi comme la composante la plus essentielle dans l'autonomie d'un robot étant donné que les décisions et les actions qu'il peut prendre en dépendent directement.

Dans sa formulation la plus simple, la perception consiste à obtenir des informations sur le monde qui entoure le robot par le biais des capteurs et construire ensuite une représentation (modèle) de cet environnement.

Problématique

Un robot qui se déplace dans un environnement inconnu pour accomplir une mission a besoin de percevoir le monde qui l'entoure et d'en faire une analyse cohérente. Le SLAM «*Simultaneous Localization And Mapping*» est une composante essentielle de la perception et sert de brique de base à une multitude d'autres tâches en robotique mobile.

En effet, il est essentiel de cerner son environnement afin de permettre au robot de raisonner et de planifier correctement des actions en fonction de ce qui l'entoure.

Dans ce mémoire, il est question de traiter la problématique du SLAM et d'effectuer plusieurs simulations de plusieurs de ces méthodes d'implémentation pour envisager de coupler un module de SLAM au module de DATMO qu'on a réalisé lors de notre projet de fin d'études

dans le but de maitre en œuvre plus tard un module de perception complet pour les robots mobiles.

Organisation du mémoire

Ce mémoire s'étend sur trois chapitres encadrés par une introduction générale et une conclusion générale.

Dans le premier chapitre, on s'intéressera au contexte général dans lequel s'insère notre travail en présentant les notions de base de la robotique mobile tout en insistant sur le lien de dépendance fort entre la navigation autonome et le SLAM.

Le second chapitre expose un état de l'art des différentes méthodes de résolution du problème de SLAM.

Le troisième chapitre a pour but de présenter l'intégralité des simulations qu'on a réalisées suivi de commentaires sur la qualité des résultats obtenus.

Chapitre 1

Généralités sur la robotique mobile

La robotique a atteint aujourd'hui son plus grand succès dans le monde de l'industrie manufacturière. Les robots manipulateurs à eux seuls représentent un marché de 2 milliards de dollars tandis que les robots mobiles connaissent un essor très important notamment les véhicules autonomes dont le marché économique semble immense et très prometteur.

1.1 Robot mobile autonome

De manière générale, on regroupe sous l'appellation *robots mobiles* l'ensemble des robots à base mobile. Le plus souvent, lorsqu'on parle de *robots mobiles*, on sous-entend *robots mobiles à roues*. Les autres types de robots mobiles sont généralement désignés par leur type de locomotion (robots mobiles à pattes, robots mobiles volants, robots mobiles marins ...).

On distingue également dans (1) deux principaux modes de fonctionnement pour un robot mobile : télé-opéré et autonome. En mode télé-opéré, une personne contrôle le robot à distance en envoyant ses ordres via une interface de commande (généralement un joystick ou un clavier) et ceux-ci sont ensuite transmis au robot via une interface de communication (réseau local, internet, satellite, etc.). Les robots sont rarement placés dans ce mode de fonctionnement et d'ailleurs, selon le niveau de télé-opération, le terme « *robot* » est plus au moins justifié.

Inversement, en mode autonome le robot doit prendre ses propres décisions. Ce qui veut dire qu'il doit être en mesure de percevoir son environnement et de savoir comment réagir en conséquence. Cela va de la localisation et la cartographie de son environnement à la planification de son parcours et la détermination des mouvements qui vont lui permettre d'atteindre son objectif.

1.2 Applications de la robotique mobile

Les véhicules autonomes (appelés aussi *véhicules intelligents*) sont une application typique du domaine de la robotique mobile. Aussi appelés *véhicules intelligents*, les véhicules autonomes

sont des robots mobiles capables de rouler automatiquement et en toute autonomie (accélérer, freiner, tourner, éviter les obstacles, etc.) dans le trafic réel et sans l'intervention d'un être humain.

L'application de systèmes d'assistance à la conduite (ou systèmes ADAS) dans les voitures représente elle aussi une application des recherches en robotique mobile. Ces systèmes sont très utilisés dans les nouvelles voitures et permettent une réduction significative des accidents de voitures causées par inattention du conducteur.

1.3 Navigation autonome des robots mobiles

Le développement des véhicules autonomes a soulevé un grand nombre de problèmes dont la plupart ne sont toujours pas résolus. Ceux-ci viennent essentiellement du fait que ces véhicules sont destinés à évoluer de manière autonome dans environnements inconnus, peu ou pas structurés. Les principaux travaux dans ce domaine portent essentiellement sur comment développer des techniques et des technologies permettant au robot mobile de « *percevoir* » plus de choses dans son environnement et de « *penser* » davantage et plus vite.

Un des problèmes les plus importants de la robotique mobile autonome est la navigation autonome. L'objectif est de développer des techniques permettant au robot mobile de se déplacer sans intervention humaine dans un environnement inconnu et en présence d'obstacles mobiles. Cela consiste en l'exécution d'un certain nombre d'actions élémentaires (déplacement, manipulation d'objets...) qui nécessite une localisation précise, ainsi que la construction d'un bon modèle de l'environnement.

La construction du modèle de l'environnement est une étape essentielle et primordiale pour une multitude de fonctions en robotique mobile. Elle est le résultat de la résolution d'un problème plus général : la perception. En effet, un véhicule observe le monde externe à l'aide de capteurs et construit un modèle interne de l'environnement extérieur. Il met à jour continuellement ce modèle en utilisant les dernières données des capteurs (2).

Dans ce cadre, la perception peut être divisée en deux étapes : la première partie, appelée SLAM (*Simultaneous Localization And Mapping*) s'intéresse à la construction d'une carte de l'environnement extérieur et à la localisation du véhicule lui-même dans cette carte, et deuxième partie traite la détection et du suivi des objets mobiles qui s'y trouvent (DATMO pour *Detection And Tracking of Moving Objects*).

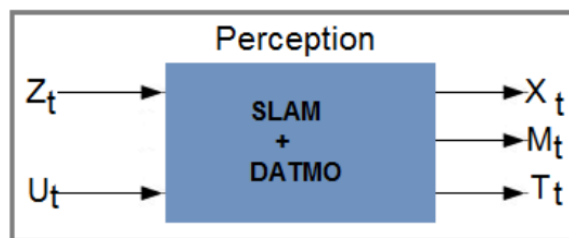


Figure 1-1: Les deux composantes de la perception : SLAM et DATMO.

1.3.1 Perception artificielle des véhicules autonomes

Littéralement la perception (du latin *perceptio*, *percipio*) est le processus de compréhension de l'environnement par l'organisation et l'interprétation des données qui proviennent des capteurs (2).

Dans le contexte de la robotique mobile (ou des véhicules intelligents), elle consiste à obtenir des informations sur l'environnement qui entoure le robot en utilisant les capteurs, un prétraitement sur les données de ces capteurs et en inférant une représentation (modélisation) de cet environnement. Concrètement, ceci est réalisé en obtenant les informations fondamentales suivantes :

- La position des objets par rapport aux autres objets. Ces informations sont appelées *carte de l'environnement* (ou *map*) ;
- La position du véhicule dans cette carte ;
- Distinction entre les objets statiques et les objets dynamiques ;
- Le type de chaque objet présent dans la carte, aussi appelé classification des informations (voiture, bicyclette, personne, ... etc.).

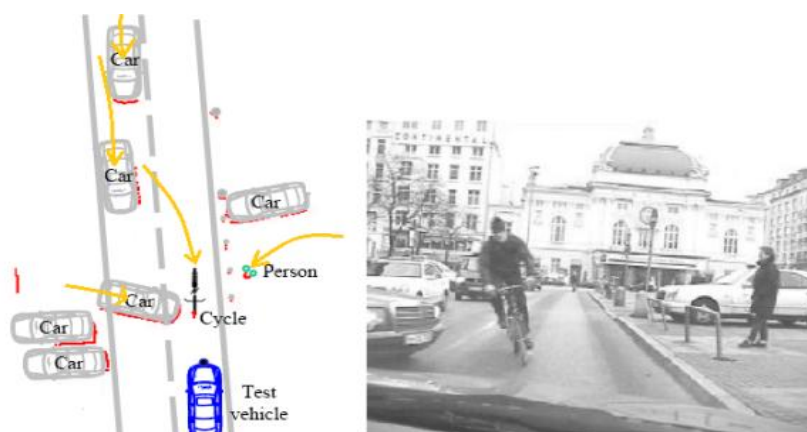


Figure 1-2 : L'objet de la perception en robotique. A droite : l'environnement réel. A gauche : le résultat de l'inférence des données capteur par le robot.

La figure ci-dessus a pour but d'illustrer l'objectif de la perception en robotique. L'image de droite montre l'environnement perçu par le robot tandis que l'image de gauche illustre la

représentation de cet environnement en inférant les données des capteurs. Les données provenant du capteur laser sont représentés par les *points rouges* et le véhicule autonome (aussi appelé *véhicule de test* ou *véhicule hôte*) est représenté en bleu. On constate d'après cette figure que le robot a inféré les formes de la route et des objets qui s'y trouvent, la position de chaque objet et distingue entre les objets statiques et les objets dynamiques (illustré par les flèches jaunes). Une telle représentation est généralement obtenue en fusionnant les données de plusieurs capteurs.

La perception est le problème le plus important à résoudre en vue de permettre au robot de fonctionner sans intervention humaine. La perception, telle que définie plus haut, requiert la résolution des cinq sous-problèmes suivants :

- *Mapping* : permet d'obtenir les relations géométriques entre les objets statiques présents dans l'environnement. La résolution de ce problème est nécessaire pour la construction d'une carte en temps réel en utilisant les données des capteurs.
- *Localisation* : définie comme étant le processus permettant d'obtenir la relation entre le véhicule autonome et les objets statiques. Un véhicule autonome doit connaître sa position dans la carte de l'environnement.
- *Détection des objets mobiles* : permet au robot de distinguer entre les objets statiques et les objets dynamiques.
- *Suivi des objets mobiles* : défini comme étant le processus permettant d'établir les relations géométriques et temporelles entre le robot autonome, les objets mobiles et l'environnement statique.
- *Classification des objets* : La résolution de ce problème permet de connaître le type de chaque objet, notamment les objets dynamiques.

Les quatre premiers sous-problèmes sont regroupés dans la littérature en deux principaux problèmes : SLAM (*Simultaneous Localization And Mapping*) et DATMO (*Detection And Tracking of Moving Objects*) (2).

La résolution du problème SLAM permet d'obtenir la carte de l'environnement où évolue le robot ainsi que la position, à chaque instant, de celui-ci dans cette carte. Tandis que le DATMO lui permet de prendre en considération le mouvement des objets mobiles de sorte à ce que les actions appropriées soient exécutées.

La classification des objets peut être considérée comme une amélioration des résultats donnés par les deux modules SLAM et DATMO. Elle est particulièrement utile et utilisée en DATMO.

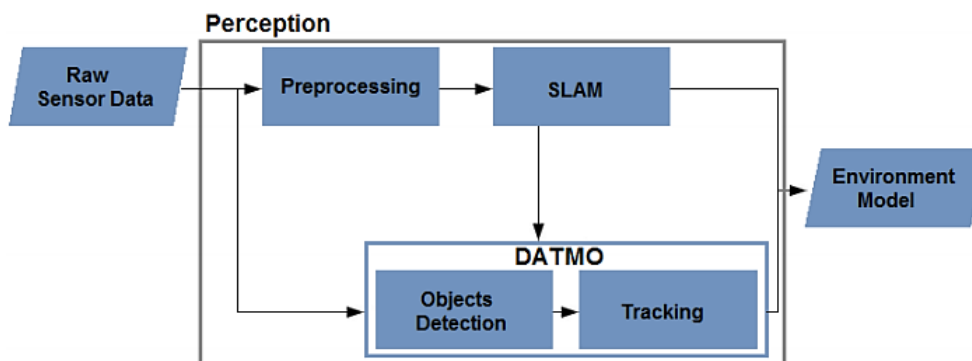


Figure 1-3 : La perception utilise les données des capteurs en entrée et produit en sortie : l'état du véhicule (position et vitesse, ...), la carte de l'environnement et une liste des objets mobiles et leurs caractéristiques cinématiques.

La carte de l'environnement incluant le comportement dynamique des objets mobiles (vitesses, accélérations, ...) donnée par la résolution du problème SLAM et DATMO est utilisée pour des applications de plus haut niveau telle que l'assistance à la conduite du véhicule. En effet, si le robot parvient à accomplir le SLAM et le DATMO d'une manière fiable et en temps réel, on peut utiliser le modèle généré pour, par exemple, détecter des situations critiques et alerter en avance le conducteur ou d'actionner automatiquement les freins en cas d'une collision inévitable.

Finalement, on peut dire que l'importance de la perception peut être vue à partir du fait qu'elle est le premier élément dans l'architecture d'un système robotique qui regroupe également la décision et l'action. Cette architecture est aussi appelée PDA (pour *perception, decision, action* en anglais) (voir la figure ci-dessous). On constate d'emblée que les actions du robot dépendent des décisions qu'il a prise, qui elles, dépendent directement du modèle de l'environnement produit par la perception.

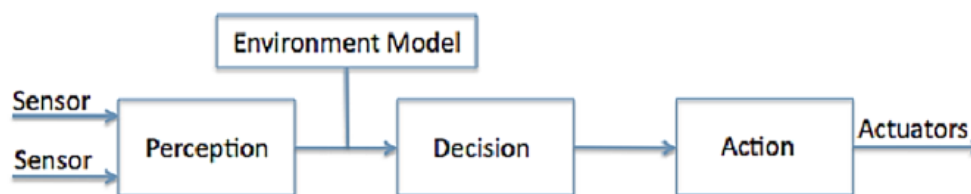


Figure 1-4 : Architecture PDA d'un système robotique

1.3.2 Présentation et définition du SLAM

Le paragraphe précédent nous a permis de conclure que les systèmes de perception artificielle sont un élément majeur dans les applications de la robotique en général. Les industriels commencent à exprimer un intérêt fort pour ces technologies.

Le SLAM consiste en la résolution du problème de Mapping par la création d'une carte au fur et à mesure que le robot évolue, ainsi qu'à la résolution du problème de *Localisation* par la localisation du véhicule ou du robot en question sur cette carte.

En SLAM, la cartographie de l'environnement et la localisation du véhicule dans cette carte sont deux problèmes rigidelement liés. En effet, on ne peut pas tracer la carte de l'environnement sans avoir la position du véhicule et on ne peut pas déduire la position du véhicule dans la carte sans avoir cette carte. Qadeer Baig a affirmé dans (2) que le SLAM est un *chicken egg problem*, sa résolution nécessite des méthodes pouvant évaluer la carte de l'environnement et la position du véhicule dans cette carte de façon conjointe.

1.4 Conclusion

A travers ce premier chapitre, nous avons présenté de façon générale la problématique du SLAM ainsi que le contexte général dans lequel elle s'insère. Nous avons notamment insisté sur la relation qui existe entre la perception, la navigation autonome et les systèmes d'assistance à la conduite.

Nous soulignons également que l'utilisation du SLAM en navigation autonome est une approche qui a été développée pendant plus qu'une décennie et de nombreuses méthodes intéressantes et efficaces ont été mises au point.

Ce premier chapitre sera complété par une étude plus détaillée du problème de cartographie et de localisation simultanée suivie d'une taxonomie des différentes classes d'approches de résolution existantes.

Chapitre 2

Etat de l'art SLAM

Le développement d'une solution au problème de localisation et de cartographie simultanée a été, pendant plus qu'une décennie, un champ de recherches stratégique dans le cadre du développement du véhicule autonome. En effet, rendre un véhicule capable de se mouvoir sans action externe requière des capacités de perception minimales dont le SLAM est la composante la plus primaire.

Dans le but de résoudre ce problème, plusieurs approches de formulation et de résolution du SLAM ont été proposées pour différents types d'environnements.

Ce deuxième chapitre a pour but de présenter les différentes méthodes développées pour la localisation et de cartographie simultanées (SLAM) en commençant par présenter les différents types de cartes utilisées pour modéliser l'environnement.

2.1 Méthode de Modélisation de l'environnement pour le SLAM

Les méthodes de modélisation de l'environnement les plus utilisées en SLAM sont classées selon (3) en deux grandes familles :

- Les méthodes de modélisation topologiques basées sur des graphes représentant des informations caractéristiques de l'environnement (coins, croisement de deux couloirs, jonctions en T, etc.).
- les méthodes de modélisation métriques qui décrivent explicitement la position « géométrique » des éléments de l'environnement.

On va à présent passer à la présentation de ces deux familles de méthodes servant à modéliser l'environnement en SLAM.

2.1.1 Méthodes de modélisation métriques de l'environnement

Cette famille de méthodes de modélisation de l'environnement peut se diviser en deux grandes catégories, les méthodes géométriques et les méthodes probabilistes basées sur le formalisme des grilles d'occupation

2.1.1.1 Méthodes géométriques

Ces méthodes décrivent explicitement les coordonnées cartésiennes des amers caractéristiques de l'environnement (Voir figure 2.1).

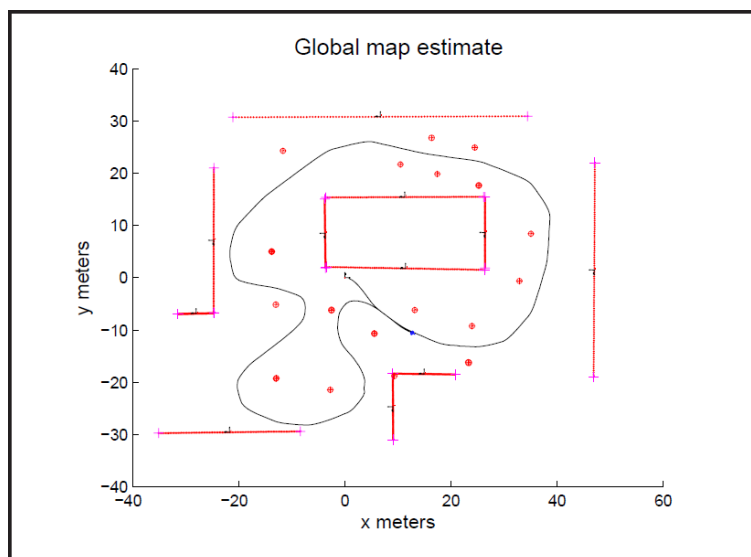


Figure 2-1: Carte géométrique de l'environnement.

Pour avoir des résultats plus précis, la forme géométrique des obstacles appartenant à l'environnement peut varier suivant leurs natures ou suivant la nature du regroupement de points les représentant (4).

Le problème relatif à l'apparition des *mouvements fantômes* est traité en attribuant à chaque forme géométrique un point appelé « ancre ». Cette méthode est détaillée dans (5).

2.1.1.2 Méthodes probabilistes

Malgré les améliorations apportées par le modèle géométrique, une représentation plus robuste des environnements encombrés où de nombreux objets y manœuvrent en même temps est nécessaire.

C'est dans cette optique là que la représentation de l'environnement par une grille d'occupation fut introduite par A. Elfes dans (6) en 1989. Cette représentation se base sur la division de la scène perçue par le robot en un ensemble de cellules de taille uniforme (souvent) ou variable. Elle attribue à chaque cellule une probabilité d'occupation notée $P(O_i)$ qui représente la probabilité qu'un objet occupe la cellule i . Cette probabilité est calculée à partir des informations fournies par les capteurs et sa valeur traduit l'état de la cellule telle que :

$$P(O_i): \begin{cases} > 0.5 & \text{si la cellule } i \text{ est occupée} \\ = 0.5 & \text{l'état de la cellule } i \text{ est inconnu} \\ < 0.5 & \text{si la cellule } i \text{ est libre} \end{cases}$$

Dans le contexte des véhicules intelligents, la grille d'occupation est la plus utilisée pour représenter l'environnement extérieur. Elle est particulièrement adaptée à la fusion des données fournies par plusieurs capteurs.

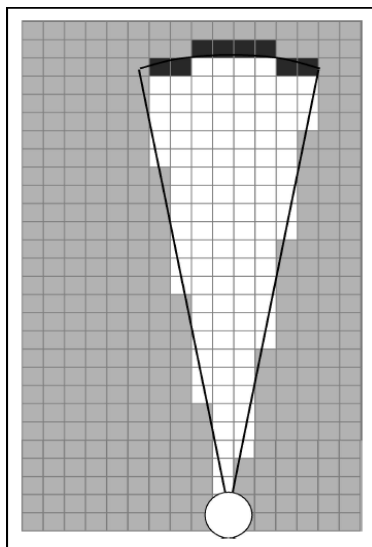


Figure 2-2: Représentation de l'environnement avec une grille d'occupation.

Cette dernière modélisation se distingue des deux modélisations précédentes par le fait que des notions telles que « objet » et « track » n'existent plus. Elles sont remplacées par la notion de cellules et on ne s'intéresse qu'à l'estimation de leur occupation et de leur vélocité en faisant abstraction de la forme des objets (obstacles) qu'elles peuvent contenir.

2.1.2 Méthodes de modélisation topologique

Le principe d'une représentation topologique de l'environnement est de décrire l'environnement par des graphes qui représentent la connectivité entre des lieux caractéristiques et prépondérants de l'environnement au sens de la navigation (3).

La figure suivante montre deux cartes différentes reflétant le même environnement perçu par le robot, les cartes de gauche sont des cartes topologiques alors que celles de droite sont des cartes métriques du type grille d'occupation.

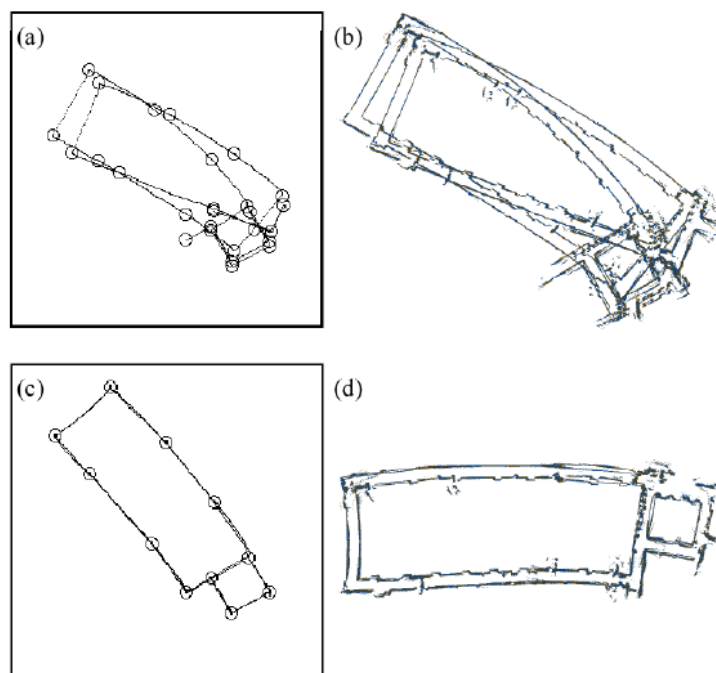


Figure 2-3: comparaison entre les deux catégories de représentation de l'environnement, A gauche : représentation topologique de l'environnement, à droite : représentation métrique

Maintenant qu'on a présenté les principales méthodes de modélisation de l'environnement utilisées en localisation et cartographie simultanée, on va passer à la présentation des principales techniques de résolution de ce problème.

2.2 Les différentes approches de cartographie et localisation simultanée (SLAM)

Etant donné l'importance du module de SLAM dans la résolution du problème de perception artificiel pour véhicules autonomes, plusieurs recherches ont été menées dans le but de concevoir une solution optimale permettant d'assurer une détection fiable. Cette partie a pour but de présenter les deux principales approches de résolution du problème de cartographie et localisation simultanée présenté par (7).

La première approche consiste en une formulation probabiliste du problème de SLAM, quant à la seconde, elle regroupe les méthodes dite de *scan-matching* qui utilise des relations géométriques entre le véhicule et ses observations.

2.2.1 Approches probabilistes

2.2.1.1 Formulation probabiliste du problème de SLAM

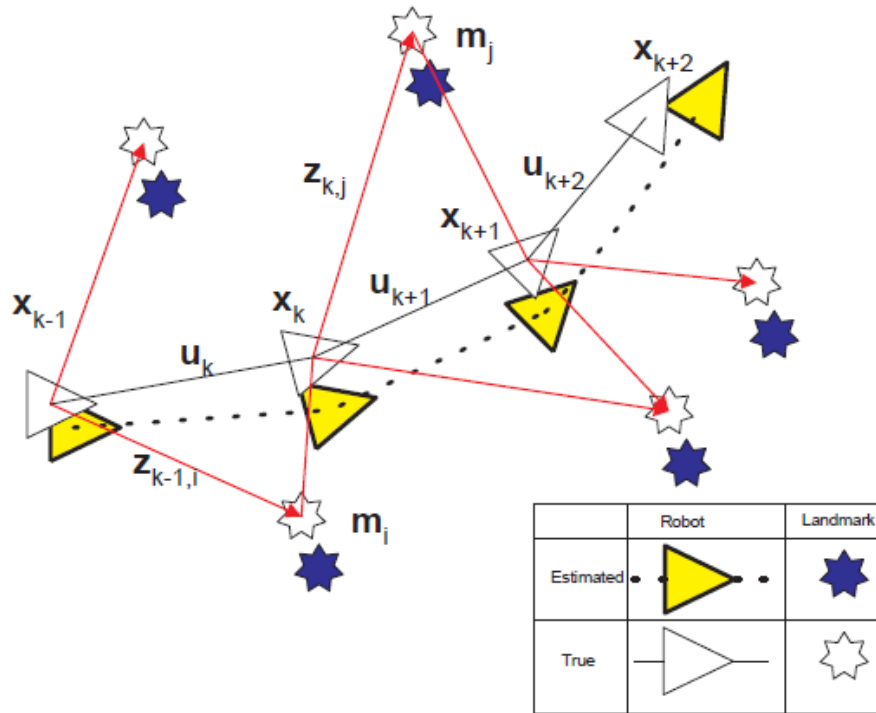


Figure 2-4: Représentation du problème de SLAM : une estimation simultanée de la position du robot et de celle des points de l'environnement.

Considérons un véhicule autonome qui se déplace dans un environnement inconnu. Tout au long de son déplacement, ce véhicule enregistre des observations en utilisant un capteur extéroceptif (voir figure 2.4). A chaque instant « k » les variables suivantes sont définies :

\mathbf{x}_k : Vecteur d'état décrivant la position et l'orientation du véhicule

\mathbf{u}_k : Vecteur de commande appliqué pour amener le véhicule à l'état \mathbf{x}_k à l'instant k

\mathbf{m}_i : Vecteur décrivant la position du $i^{\text{ème}}$ point de l'environnement

\mathbf{z}_k : Vecteur d'observation du véhicule à l'instant k

Les notations suivantes sont également utilisées

$\mathbf{x}_{0:k} = \{\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_k\}$: historique des états du véhicule.

$\mathbf{u}_{0:k} = \{\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_k\}$: historique des états de commande.

$\mathbf{m} = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_n\}$: l'intégralité des points appartenants à l'environnement

$\mathbf{z}_{0:k} = \{\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_k\}$: historique des observation

Dans sa forme probabiliste, la résolution du problème de SLAM consiste en la détermination de la densité de probabilité conjointe suivante

$$P(\mathbf{x}_k, \mathbf{m} \mid \mathbf{Z}_{0:k}, \mathbf{U}_{0:k}, \mathbf{x}_0)$$

Cette densité de probabilité représente la densité conjointe a posteriori de l'ensemble des points de l'environnement (décrivant la carte de l'environnement) et de l'état du véhicule a l'instant « k », sachant l'historique des observations.

En générale, une solution récursive est adoptée pour résoudre ce problème de SLAM en temps réel, la forme de cette solution est la suivante :

$$P(\mathbf{x}_k, \mathbf{m} \mid \mathbf{Z}_{0:k}, \mathbf{U}_{0:k}, \mathbf{x}_0) = \frac{P(\mathbf{z}_k \mid \mathbf{x}_k, \mathbf{m})P(\mathbf{x}_k, \mathbf{m} \mid \mathbf{Z}_{0:k-1}, \mathbf{U}_{0:k}, \mathbf{x}_0)}{P(\mathbf{z}_k \mid \mathbf{Z}_{0:k-1}, \mathbf{U}_{0:k})}$$

Avec :

$$P(\mathbf{z}_k \mid \mathbf{x}_k, \mathbf{m}).$$

Représente le modèle probabiliste du capteur, il décrit la probabilité d'avoir une observation \mathbf{z}_k , lorsque la position du véhicule et la carte de l'environnement sont connus.

$$P(\mathbf{x}_k, \mathbf{m} \mid \mathbf{Z}_{0:k-1}, \mathbf{U}_{0:k}, \mathbf{x}_0) = \int P(\mathbf{x}_k \mid \mathbf{x}_{k-1}, \mathbf{u}_k) \times P(\mathbf{x}_{k-1}, \mathbf{m} \mid \mathbf{Z}_{0:k-1}, \mathbf{U}_{0:k-1}, \mathbf{x}_0) d\mathbf{x}_{k-1}$$

Avec

$$P(\mathbf{x}_k \mid \mathbf{x}_{k-1}, \mathbf{u}_k)$$

Représentant le modèle probabiliste de mouvement du véhicule autonome.

La résolution du problème de SLAM consistera donc dans sa formulation probabiliste à déterminer de façon récursive la densité à posteriori de probabilité conjointe. Plusieurs méthodes ont été développées, on va présenter les plus utilisées.

2.2.1.2 Solutions au problème probabiliste de SLAM

Les solutions probabilistes du problème de SLAM consistent en une formulation appropriée du modèle du capteur et du modèle de mouvement du véhicule. Ces formulations permettront un calcul efficace de la densité a posteriori défini dans la partie précédente.

SLAM par filtrage de Kalman

La base d'une solution probabiliste du problème de SLAM en utilisant le filtre de Kalman est une description du modèle du capteur et du modèle de mouvement du véhicule de la façon suivante (8)

$$P(\mathbf{x}_k \mid \mathbf{x}_{k-1}, \mathbf{u}_k) \iff \mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k) + \mathbf{w}_k,$$

$$P(\mathbf{z}_k \mid \mathbf{x}_k, \mathbf{m}) \iff \mathbf{z}(k) = \mathbf{h}(\mathbf{x}_k, \mathbf{m}) + \mathbf{v}_k,$$

Avec « f » représentant le modèle cinématique du véhicule et « h » le modèle d'observation. W_k et V_k sont des erreurs gaussiennes à moyenne nul dont les covariances sont Q_k et R_k respectivement.

On utilisera ensuite un filtrage de Kalman pour déterminer les états du véhicule et la position des points de l'environnement ainsi que la distribution a posteriori les représentants. Plusieurs variantes du filtre de Kalman peuvent être utilisées, par exemple la solution la plus naturelle est de linéariser les fonctions « f » et « h » pour appliquer un filtrage de Kalman étendu (8). Cependant, et vu que la linéarisation effectuée sur les fonctions « f » et « h » peuvent faire diverger l'algorithme de SLAM. Une solution est l'utilisation du filtre de Kalman inodore (9). Ce filtre permet de décrire l'approximation gaussienne par un nuage de points pondérés, mis à jour récursivement en utilisant les modèles non linéaires (10). Ainsi, le calcul des deux premiers moments sera effectué sur ce nuage de points et ne nécessitera plus de linéarisation.

Les figures suivantes présenteront les résultats de l'application des méthodes EKF-SLAM et UKF-SLAM pour résoudre le problème de cartographie et localisation simultanée.

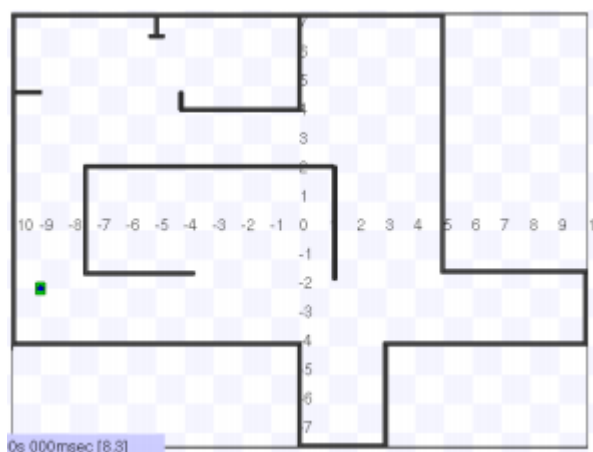


Figure 2-5 : Environnement réel

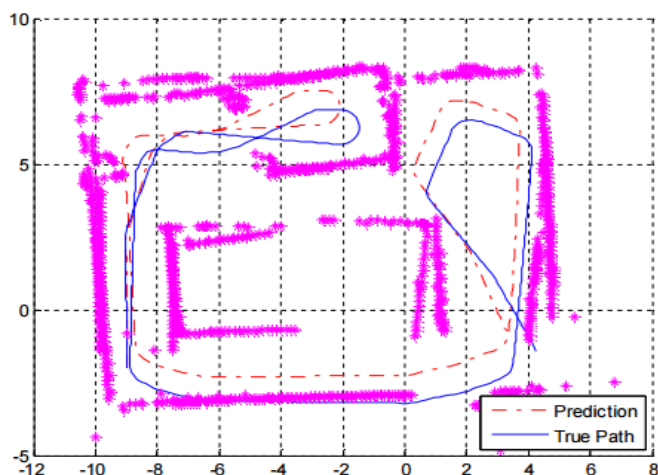


Figure 2-6 : EKF SLAM

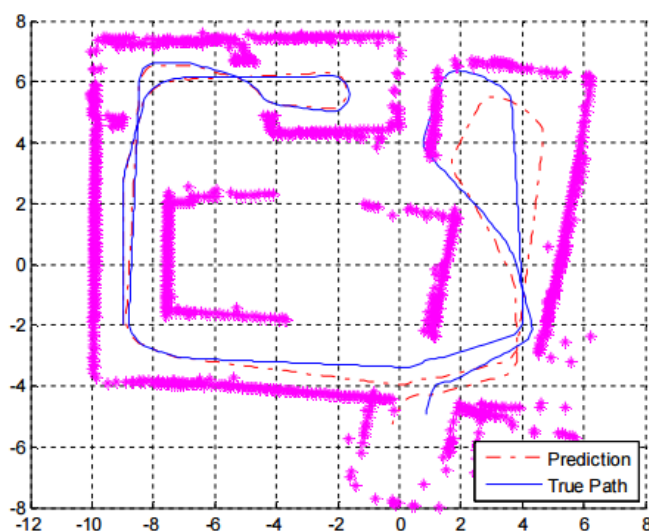


Figure 2-7 : UKF SLAM

Une comparaison entre ces deux méthodes peut être trouvée dans (9).

FAST SLAM

La méthode Fast-SLAM, introduite par (11), a marqué une nouvelle aire dans la résolution du problème d'un SLAM probabiliste récursif.

Au lieu d'essayer d'améliorer les performances de l'EKF-SLAM, cette méthode se base sur un échantillonnage de Monte Carlo pour représenter directement les non linéarités des modèles au lieu de traiter des modèles linéarisés. Cependant, la taille des environnements que traite le problème de SLAM étant très grande, l'utilisation du Rao-Blackwellisation devient très vite indispensable pour réduire l'espace d'échantillonnage.

La figure suivante représente le résultat d'un module FAST SLAM

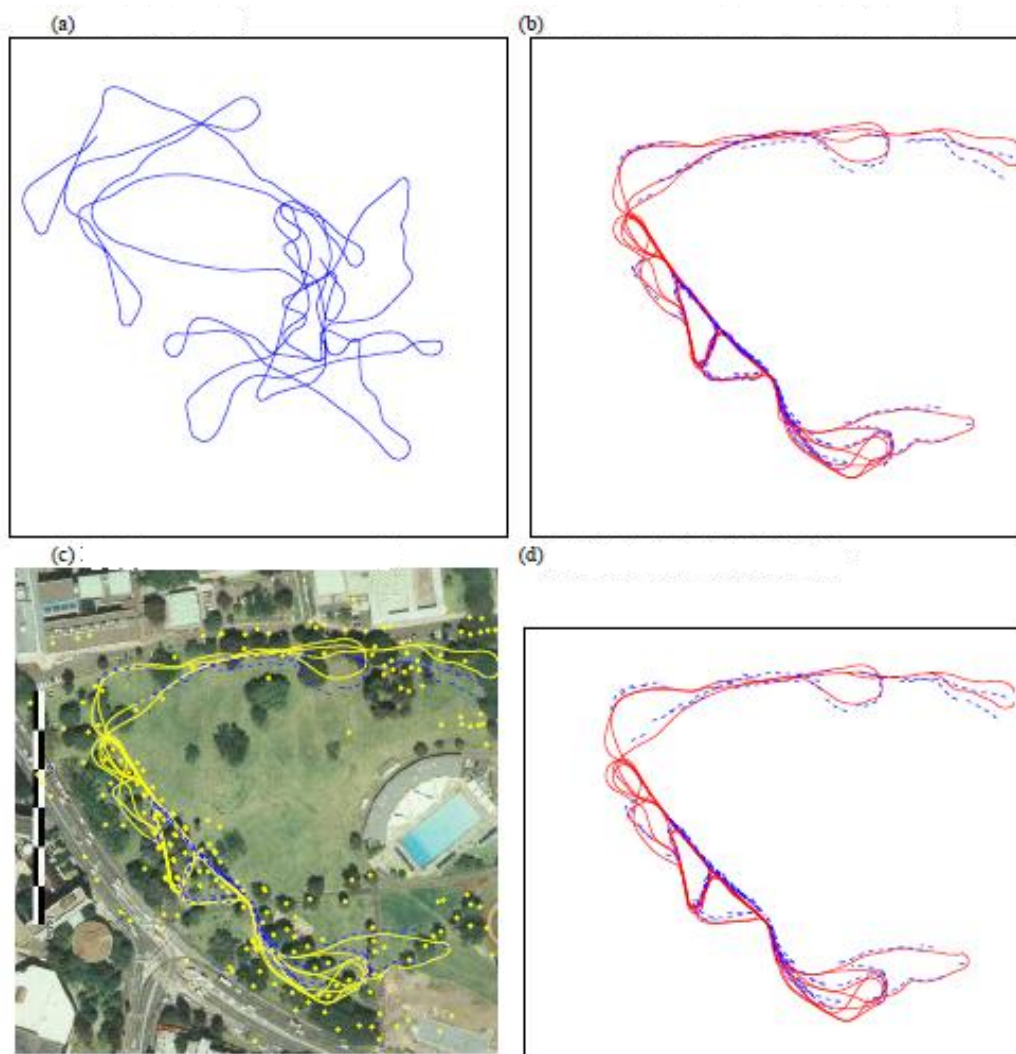


Figure 2-8 : (a) trajectoire du véhicule prédite par odométrie, (b) Trajectoire réelle (en pointillée) et trajectoire à partir de l'algorithme Fast-SLAM (en rouge), (c) résultats de l'application du Fast-SLAM en environnement externe

Il est à noter que l'estimation de la position du véhicule en utilisant la trajectoire déduite de l'odométrie est fortement déconseillée vu que cette trajectoire est déterminée par intégration du déplacement du robot à partir de ses codeurs incrémentales. Cette trajectoire est donc très sensible aux erreurs d'intégration et devient après peu de temps inutilisable ou complètement erronée.

On va à présent illustrer l'amélioration apportée par la méthode Fast-SLAM en la comparant à l'EKF-SLAM, lors de la présence d'erreurs dans l'odométrie du véhicule.

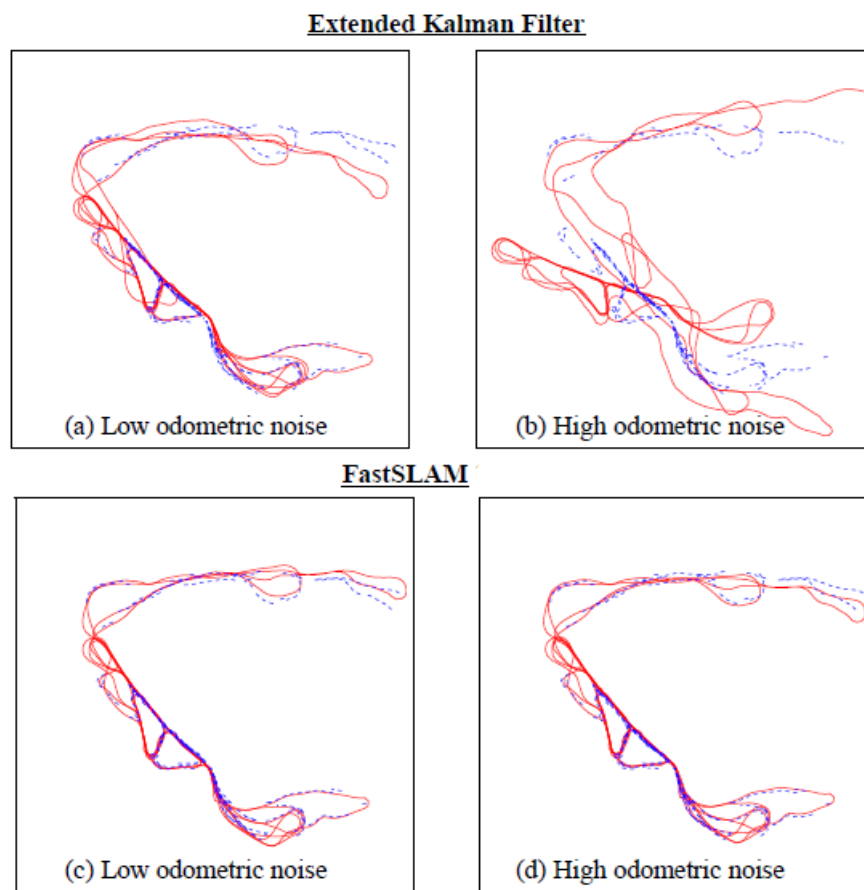


Figure 2-9 : robustesse du Fast-SLAM et de l'EKF-SLAM aux erreurs de l'odométrie.

On voit bien que la méthode FAST-SLAM est très robuste aux erreurs de l'odométrie, ceci est dû à l'utilisation d'un modèle cinématique du véhicule non linéarisée, contrairement à la méthode EKF-SLAM.

DP-SLAM

L'utilisation d'un algorithme Fast-SLAM est très efficace lorsque le nombre de points représentant l'environnement est suffisant. Cependant, dans le cas où l'environnement ne procède pas assez de points pouvant le caractériser, l'utilisation de cette méthode donnera des résultats erronés.

La méthode DP-SLAM a été introduite par (12), elle est appliquée exclusivement lors de l'utilisation d'un télémètre laser. Cette méthode n'utilise pas des points pour représenter l'environnement mais elle stock toutes les cartes possibles obtenus à partir des données laser puis utilise une nouvelle représentation pour résoudre le problème de SLAM appelée « distributed particle (DP) mapping ».

On va à présent présenter l'efficacité de cette méthode dans le cas où le véhicule autonome est amené à boucler sur un même environnement.

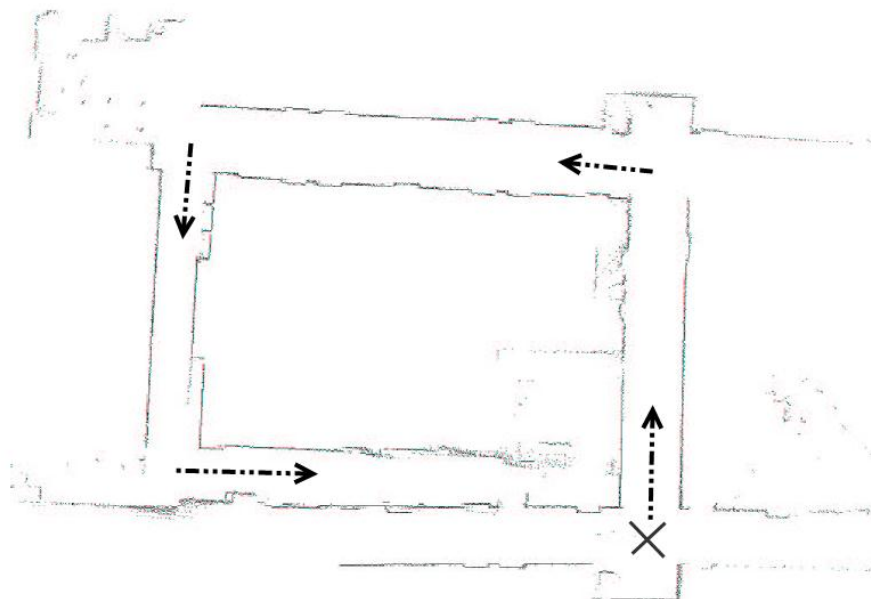


Figure 2-10 : SLAM en utilisant une seule carte

On voit bien que lors du passage du robot par son point de départ, l'algorithme de SLAM n'a pas su fusionner les résultats de la cartographie de l'état initial à l'état final. Ce problème très difficile à résoudre pour les méthodes de SLAM classiques nécessite un traitement supplémentaire qui est le plus souvent couteux en terme de temps de calculs.

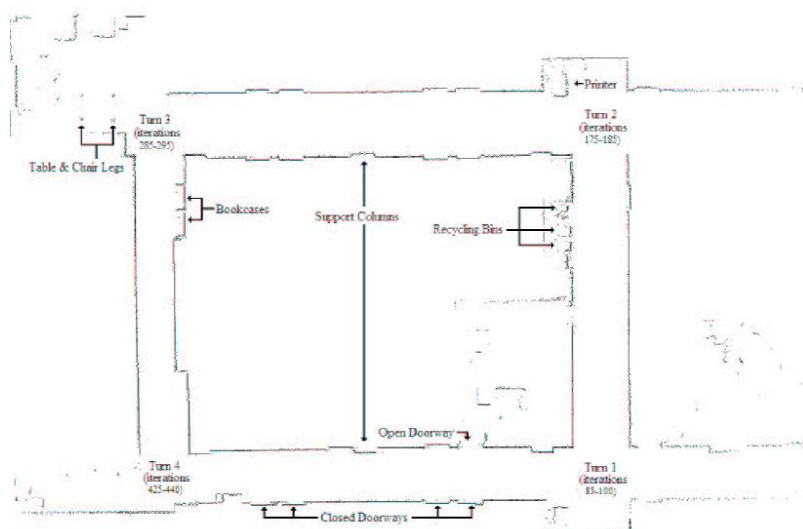


Figure 2-11 : DP-SLAM

L'utilisation de la méthode DP-SLAM a permis de corriger le problème de bouclage d'environnement par corrélation de l'historique des cartes que la méthode DP-SLAM garde en mémoire. L'utilisation d'un historique de cartes est donc une méthode efficace pour résoudre le problème de localisation et de cartographie simultanée.

On va passer à présent à la présentation des méthodes dites de « scan-matching » utilisées pour résoudre le problème de SLAM.

2.2.2 Scan-matching SLAM

Ces méthodes consistent en la mise en correspondance des scans par un processus de translations et de rotations. Cette partie aura pour but de présenter les méthodes principales développées pour résoudre le problème scan-matching SLAM.

2.2.2.1 Feature to Feature SLAM (F2F-SLAM)

Cette méthode a été développée par (13) et (14) dans le but de résoudre de façon rapide le problème de localisation et de cartographie simultanée, elle consiste à mettre en correspondance les endroits caractéristiques de l'environnement perçu par le robot (coins, cercles, etc.) elle évite donc de traiter tous les points de l'environnement ce qui rend cette méthode particulièrement rapide.

Des méthodes d'extraction de lignes, cercles et autres formes géométriques sont donc appliquées aux données des capteurs extéroceptifs pour extraire les formes qui caractérisent l'environnement. Ces formes seront ensuite traitées par l'algorithme F2F-SLAM pour modéliser l'environnement.

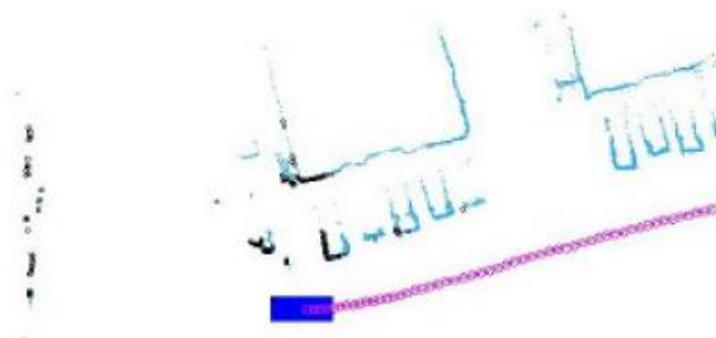


Figure 2-12 : Résultat de la méthode F2F-SLAM

2.2.2.2 Point to Feature (P2F-SLAM)

Cette méthode a été proposée par (15), elle repose sur une mise en correspondance des données capteurs avec les endroits caractéristiques de l'environnement modélisé au préalable. Son principe de fonctionnement est le suivant

- Pour chaque donnée capteur (point), on détermine l'endroit caractéristique (feature) du modèle de l'environnement la plus proche de cette donnée.
- On trouvera ensuite la congruence qui minimise la distance au carré entre les données du capteur et les endroits caractéristiques de l'environnement.
- On déplacera en suite les données capteur suivant la congruence trouvée dans l'étape précédente

- On répète les trois premières étapes jusqu'à avoir une convergence de l'algorithme.

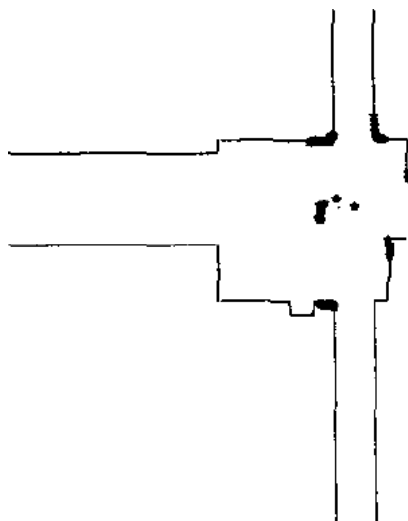


Figure 2-13 : Résultat de la méthode P2F-SLAM

2.2.2.3 Point to Point (P2P-SLAM)

Les approches basées sur l'extraction d'endroits caractéristiques de l'environnement (features) essaient d'utiliser le minimum d'informations dans la résolution du problème de SLAM pour rendre l'algorithme utilisé plus rapide. Cependant, si ces endroits caractéristiques ne sont pas détectés de façon robuste, la performance de toute la méthode de SLAM diminuera et c'est pour palier à cette dépendance que les méthodes P2P ont été développées.

Les méthodes P2P n'ont donc pas besoins d'extraction d'endroits caractéristiques de l'environnement mais ils utilisent l'intégralité des données fournis par les capteurs extéroceptifs du véhicule.

Plusieurs algorithmes P2P ont été développés, le plus utilisé est l'algorithme ICP-SLAM (ICP pour Iterative Closest Point ou Iterative Corresponding Point). Cet algorithme permet de mettre en correspondance les différents nuages de points obtenus à partir des capteurs du véhicule pour former une carte de l'environnement du robot.

Cette mise en correspondance se fait par la détermination de l'ensemble de transformations (rotation, translation) optimales qui minimise la distance entre les différents nuages de points à des instants différents.

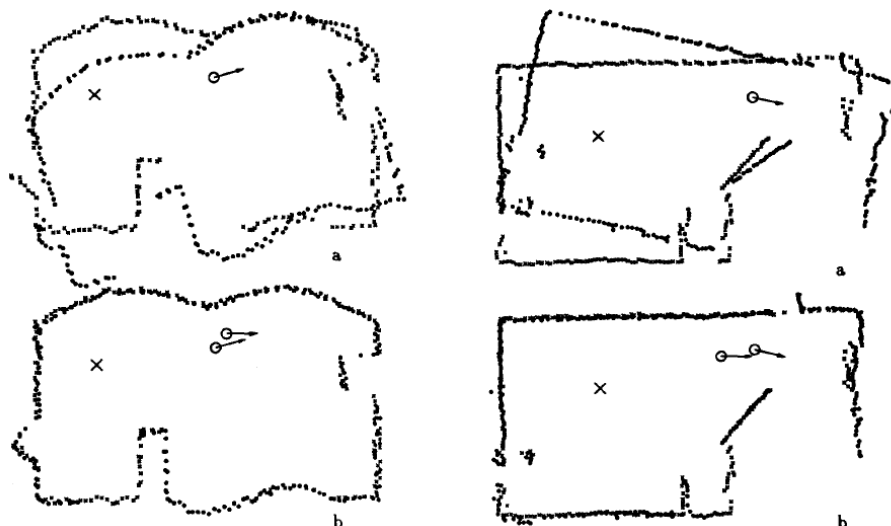


Figure 2-14 : Deux scans avant et après la mise en correspondance en utilisant la méthode ICP-SLAM

2.3 Conclusion

Ce chapitre a permis de présenter les principales méthodes utilisées pour résoudre le problème de cartographie et de localisation simultanée.

Les deux branches de méthodes existantes pour résoudre ce problème sont les méthodes probabilistes et les méthodes dites de *scan-matching*. Les méthodes probabilistes utilisent pour la formulation du problème de SLAM la théorie des probabilités, quant aux méthodes dites de *scan-matching*, elles utilisent directement les données des capteurs extéroceptifs du robot pour créer une carte de l'environnement du robot.

Chapitre 3

Simulation de méthodes *de SLAM*

Dans ce chapitre, nous nous intéresserons à la simulation de quelques méthodes de SLAM, on mettra en évidence les points forts et les points faibles de chaque méthode. Les environnements logiciels utilisés pour ces simulations sont le système d'exploitation pour robots ROS, et le logiciel MATLAB.

3.1 Simulation sous ROS

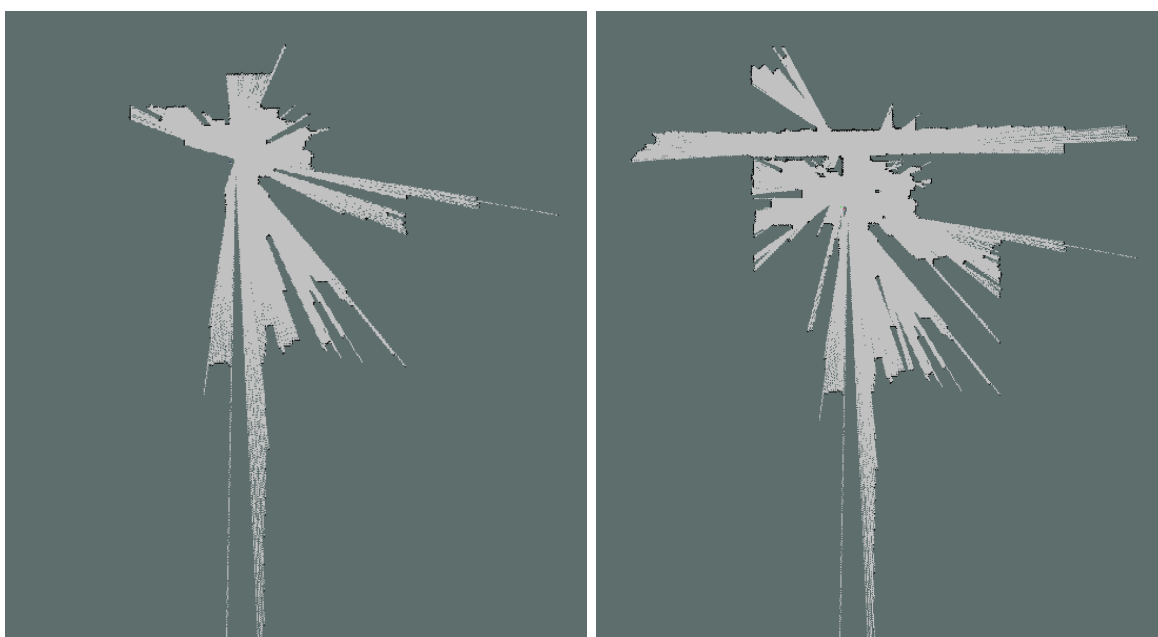
Dans cette partie, on va présenter les résultats de simulation d'une méthode probabiliste de résolution du problème de SLAM utilisant une modélisation de l'environnement sous la forme d'une grille d'occupation.

Cette méthode crée une carte de l'environnement sous la forme d'une grille d'occupation, le problème de localisation du robot est résolu en utilisant les données de l'odométrie.

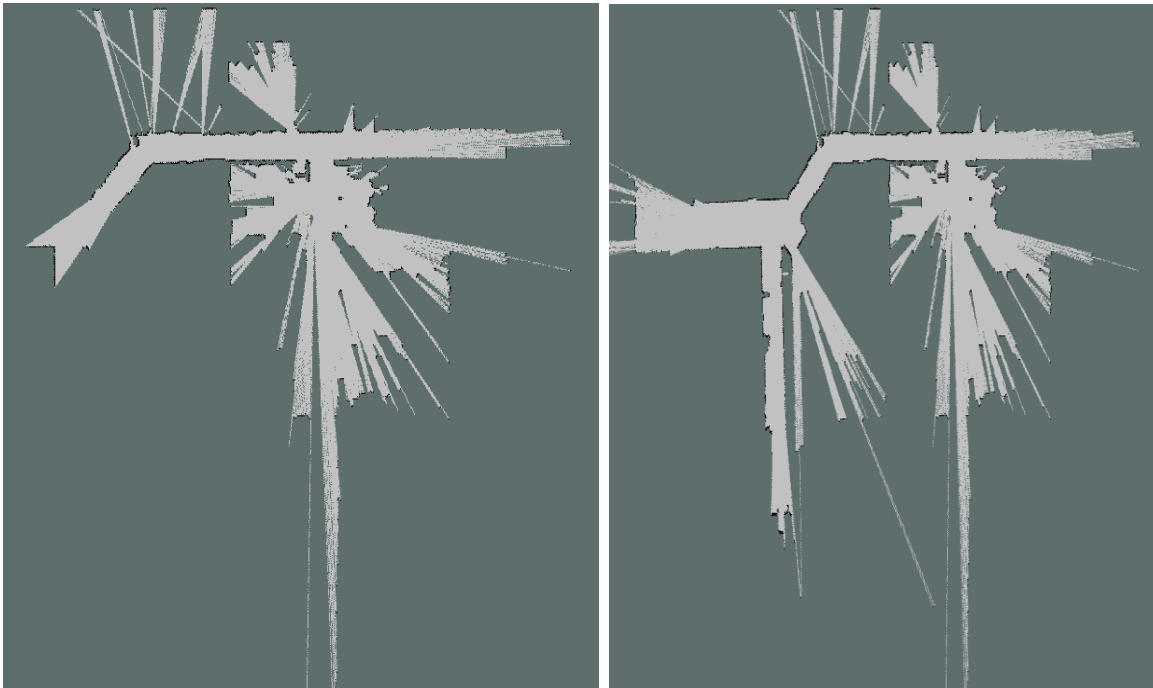
Cette simulation est réalisée à l'aide du programme (appelé *package* par la communauté de ROS) GMAPPING en utilisant le système d'exploitation pour robots ROS (16).

Résultats de la simulation

Instants t_0 et t_1 :



Instants t_2 et t_3 :



Instants t_4 et t_5 :



Les résultats que donne cette méthode sont satisfaisants, cependant dans le cas où l'environnement ou évolue le robot est large, les erreurs dus à l'odométrie deviendront très vite non tolérables et induiront en erreur ce module de SLAM. Des corrections de la position du robot sont envisageables mais ils restent non utilisables dans le cas où les applications que doit effectuer le robot exigent une grande précision.

3.2 Simulations sous MATLAB

Pour cette partie, on va présenter les résultats de la simulation de la méthode EKF SLAM et UKF SLAM ainsi que les simulations de la méthode de scan matching ICP-SLAM implémentées sous le logiciel MATLAB. La première méthode utilise l'odométrie du robot pour localiser ce dernier dans la carte de l'environnement quant à la deuxième, elle utilise l'algorithme ICP-SLAM sur un environnement statique pour la localisation.

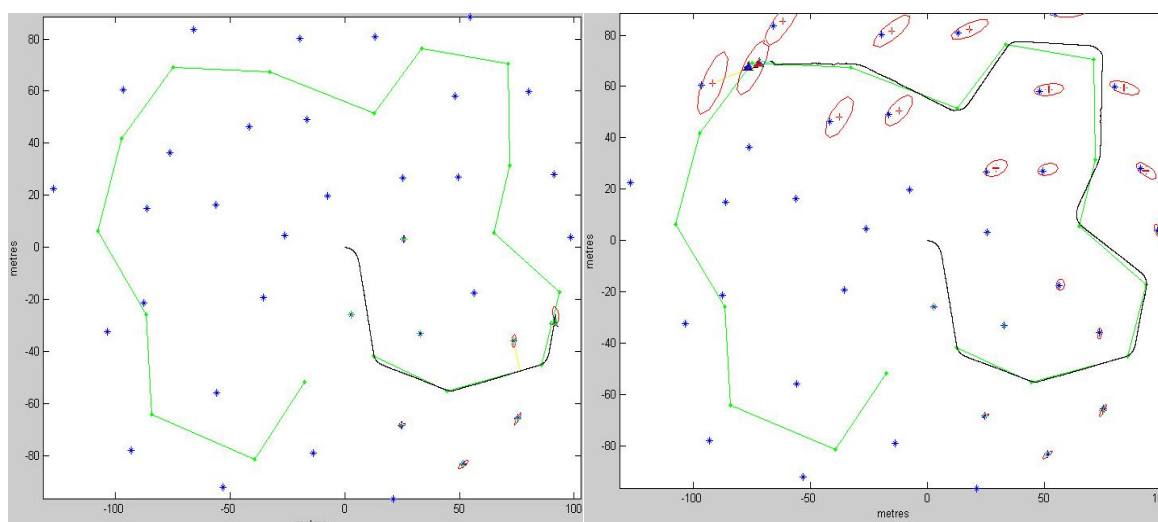
Il est à noter que la simulation des méthodes EKF et UKF SLAM n'illustre pas une reconstitution de l'environnement perçu par le robot mais elle se base sur la reconstitution de points caractérisant cet environnement.

3.2.1 Simulation de la méthode EKF-SLAM

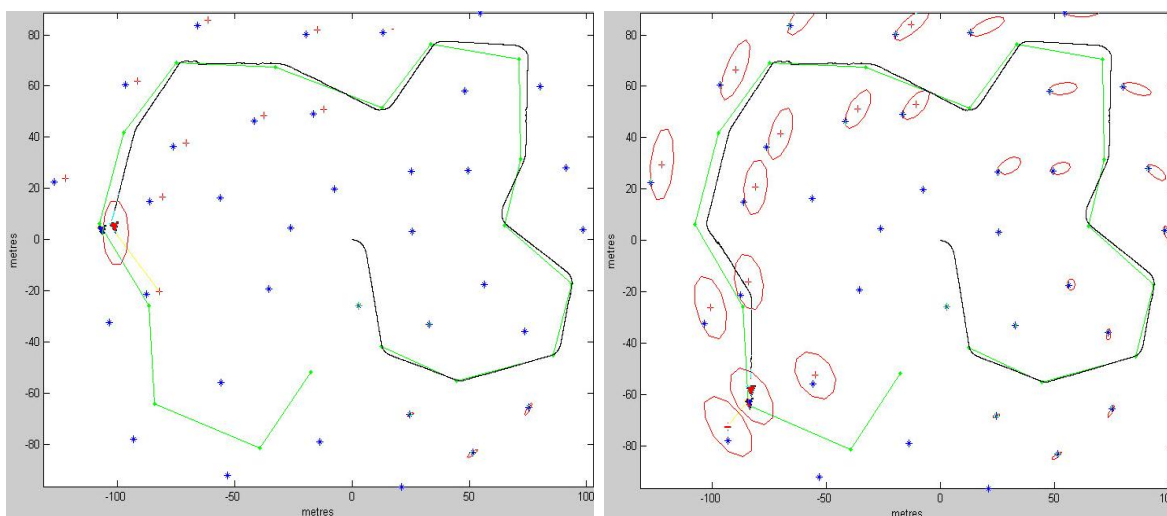
La trajectoire verte est la trajectoire réelle parcourue par le robot, la trajectoire en noir est la trajectoire estimée. Les points bleus représentent les points caractérisant l'environnement et les points rouges les points estimés. Les cercles rouges représentent l'erreur commise lors de l'estimation de la carte de l'environnement.

Résultats de la simulation

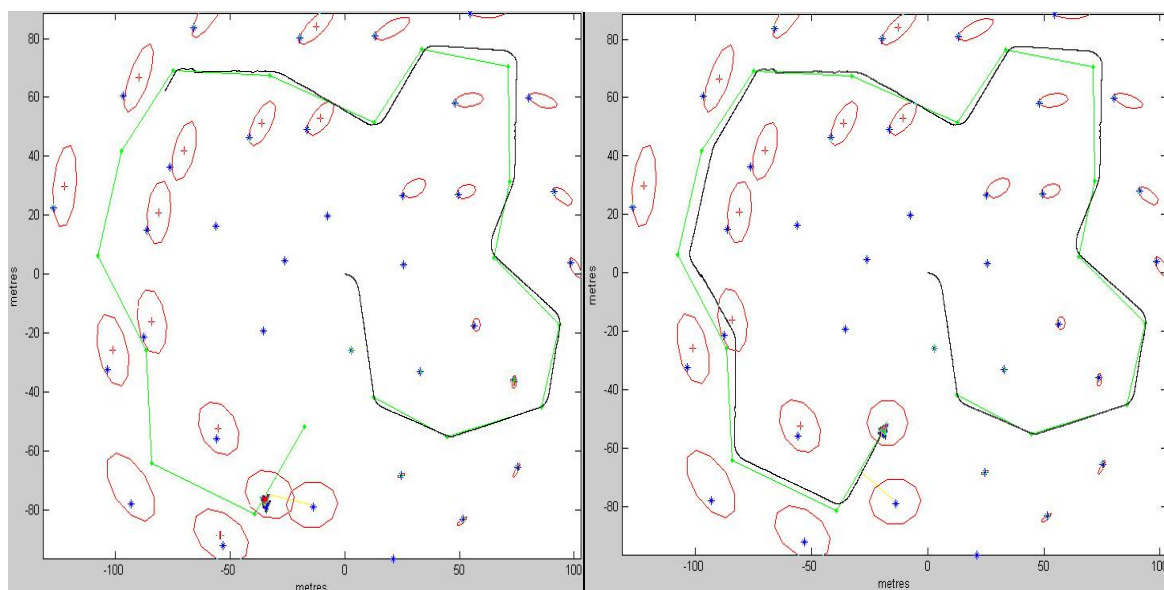
Instants t_0 et t_1 :



Instants t_2 et t_3 :



Instants t_4 et t_5 :

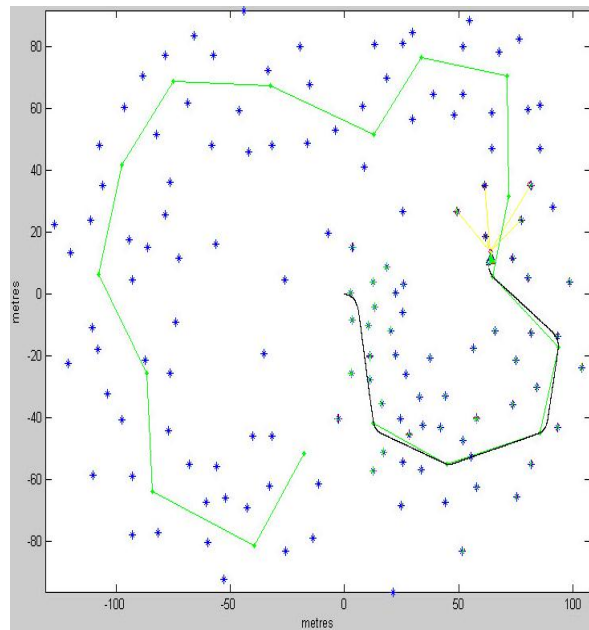
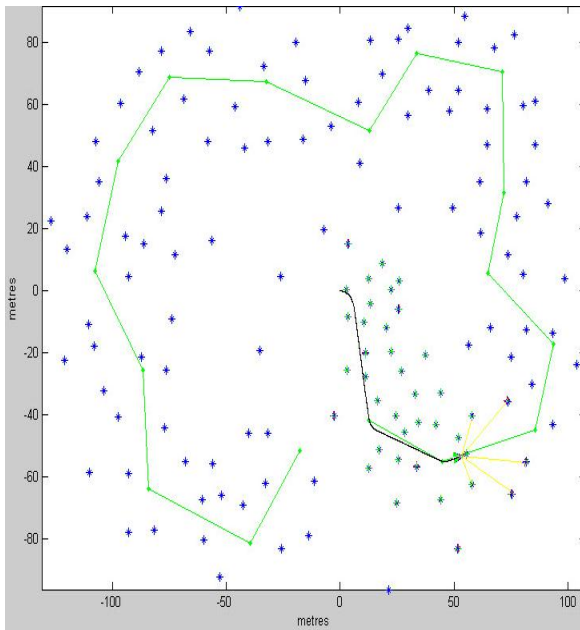


On voit que les résultats de cette simulation sont satisfaisants vu que le robot réussit à valider la plus part des points caractéristiques de l'environnement tout en estimant sa position de façon correcte. Cependant, il est à noter que dans cette simulation le problème d'association des données est considéré comme étant résolu ce qui permet une constitution de la carte de façon très robuste. Dans la réalité ce problème est loin d'être résolu et il nécessite un traitement très complexe.

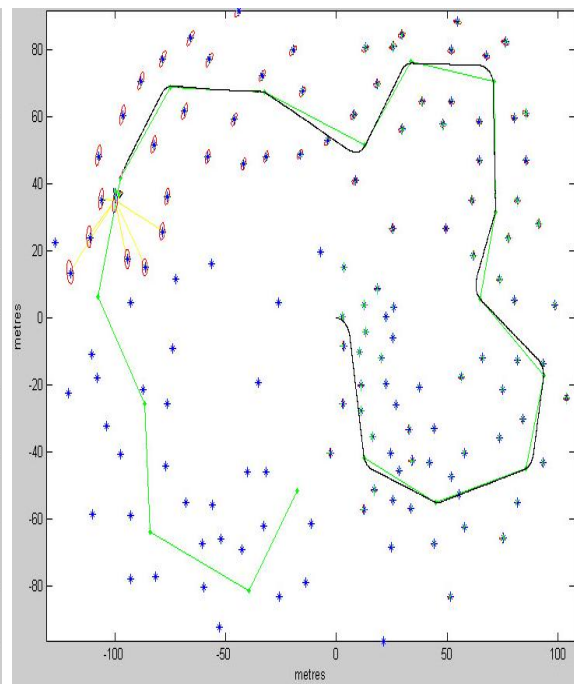
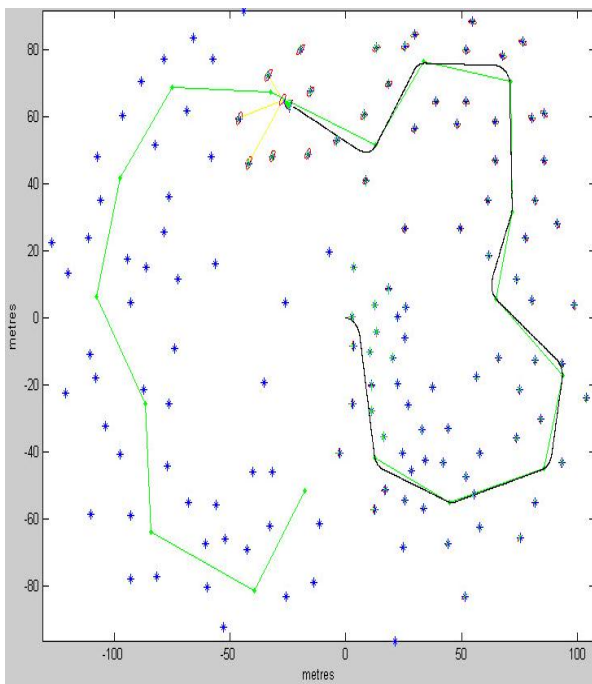
On va à présent augmenter le nombre de points caractéristiques et utiliser encore une fois l'algorithme EKF-SLAM.

Résultats de la simulation

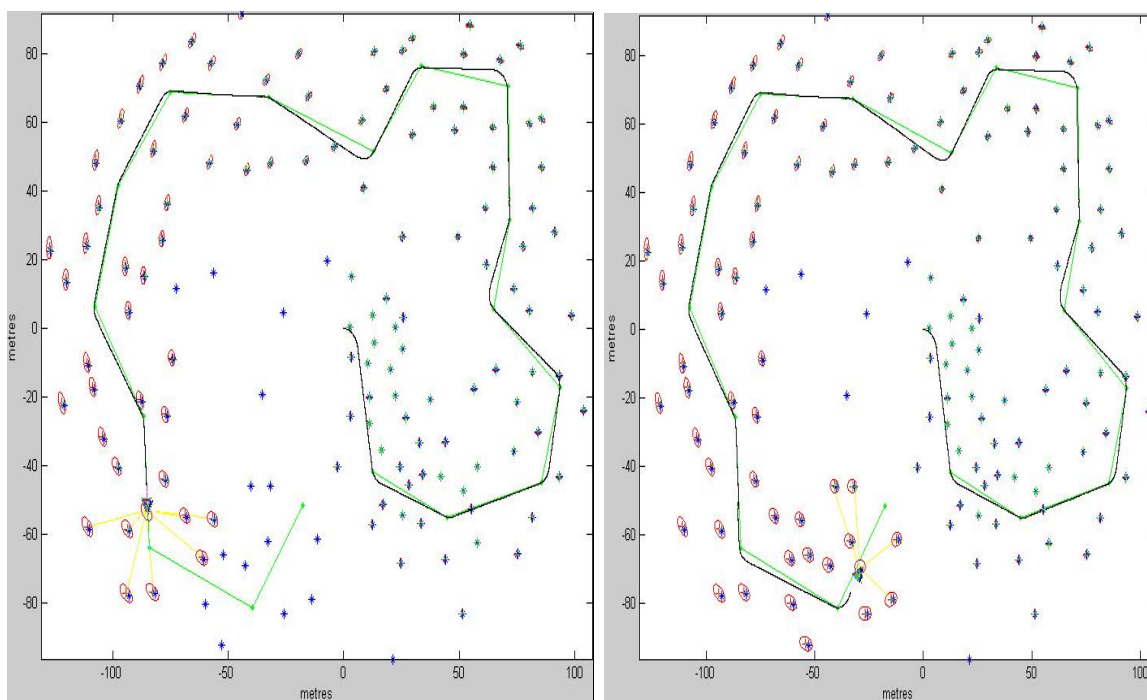
Instants t_0 et t_1 :



Instants t_2 et t_3 :



Instants t4 et t5 :

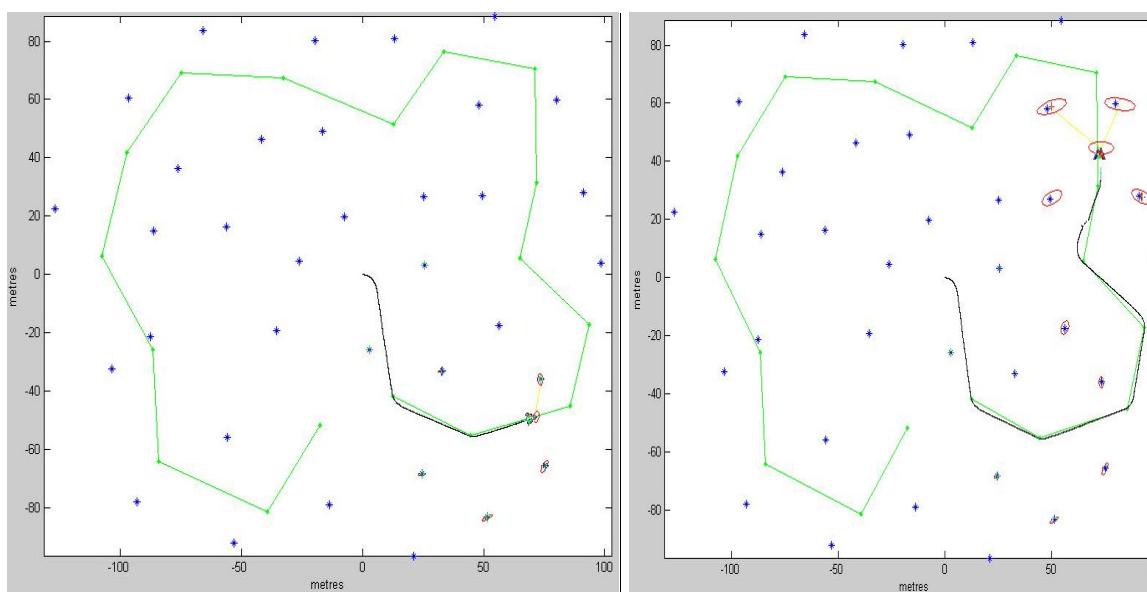


En augmentant le nombre de points, les résultats de l'algorithme EKF-SLAM sont nettement meilleurs, on voit bien que l'estimation de la position du robot s'est améliorée et que la reconstitution de l'environnement est plus correcte.

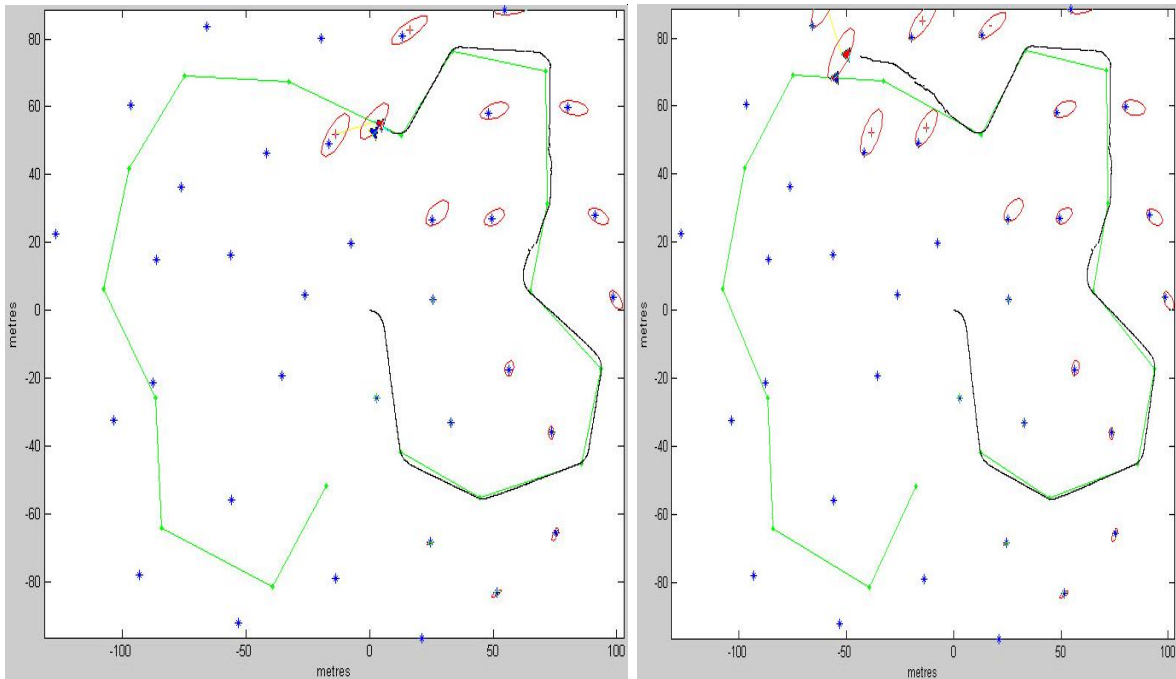
On va à présent augmenter l'erreur maximale qu'on peut avoir sur le modèle cinématique du robot à 1m/s, les résultats sont les suivants.

Résultats de la simulation

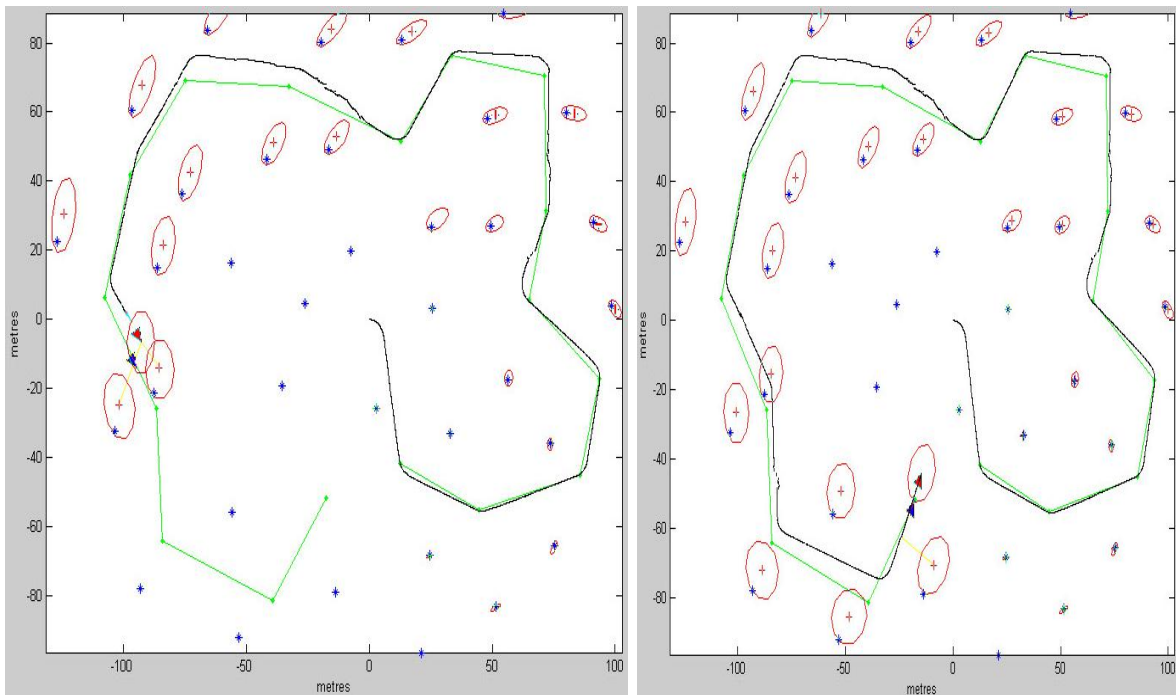
Instants t0 et t1 :



Instants t2 et t3 :



Instants t4 et t5 :

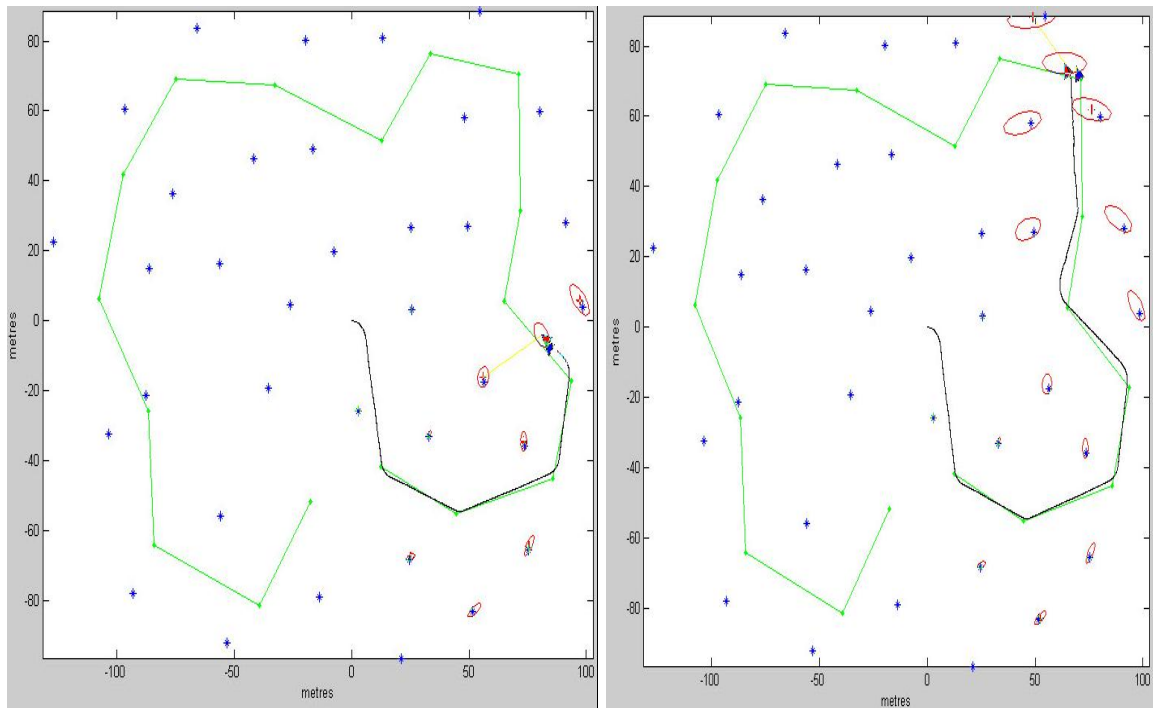


On voit bien une dégradation des résultats de l'algorithme EKF-SLAM, ceci est dû au fait que la linéarisation du modèle cinématique de déplacement du robot fait en sorte que la robustesse de cet algorithme face aux fortes perturbations diminue.

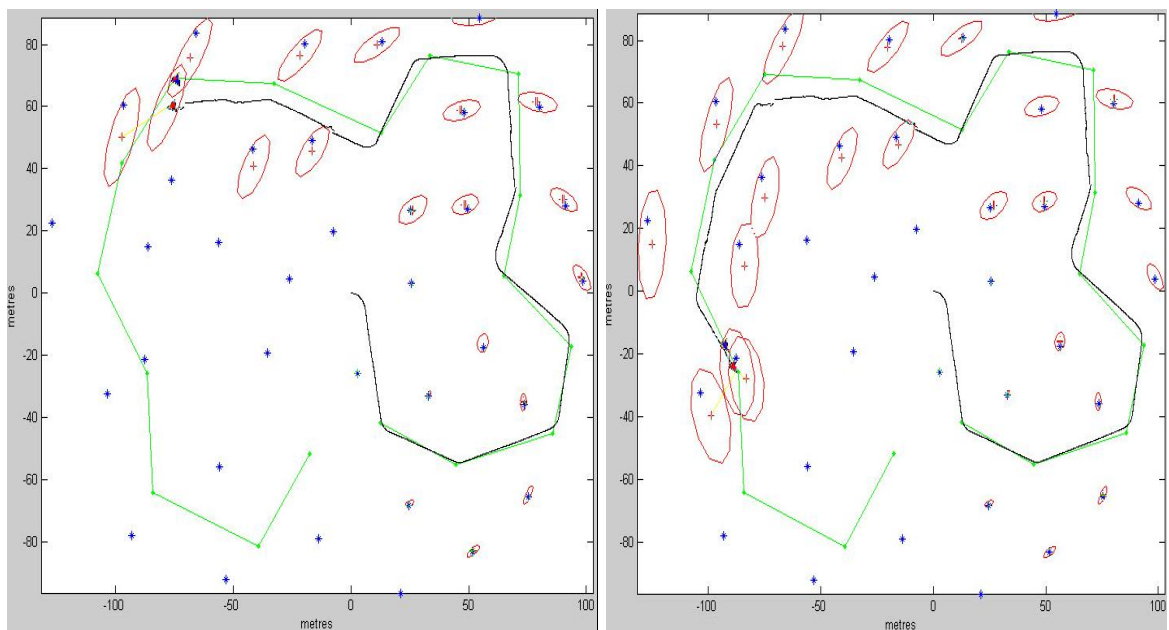
Cette dernière simulation a pour but de soumettre l'algorithme EKF-SLAM à des fortes perturbations. Pour ce faire, on va appliquer la même perturbation que précédemment sur le modèle cinématique du robot et on va introduire une perturbation de 1m sur le modèle d'observations. Les résultats sont les suivants.

Résultats de la simulation

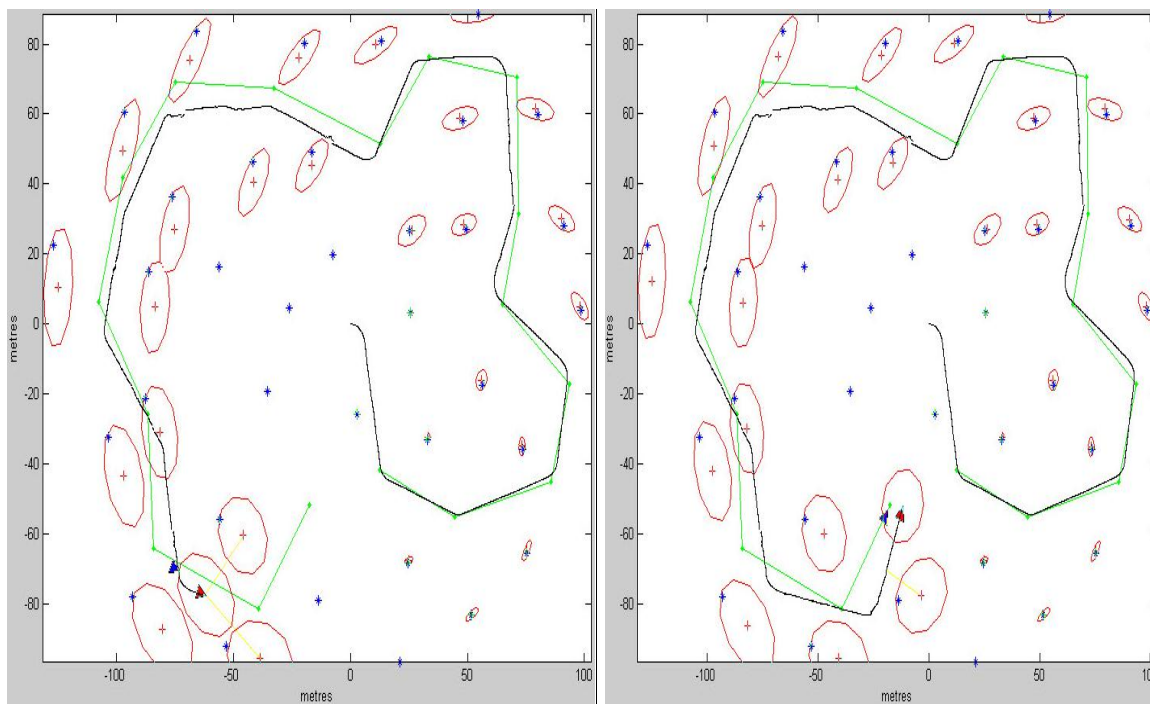
Instants t_0 et t_1 :



Instants t_2 et t_3 :



Instants t_4 et t_5 :



Encore une fois, on voit bien une dégradation des qualités du module EKF-SLAM que ce soit dans la cartographie ou dans la localisation. Ceci est dû encore une fois à la linéarisation du modèle cinématique et du modèle d'observation de ce module.

Conclusion sur la méthode EKF-SLAM

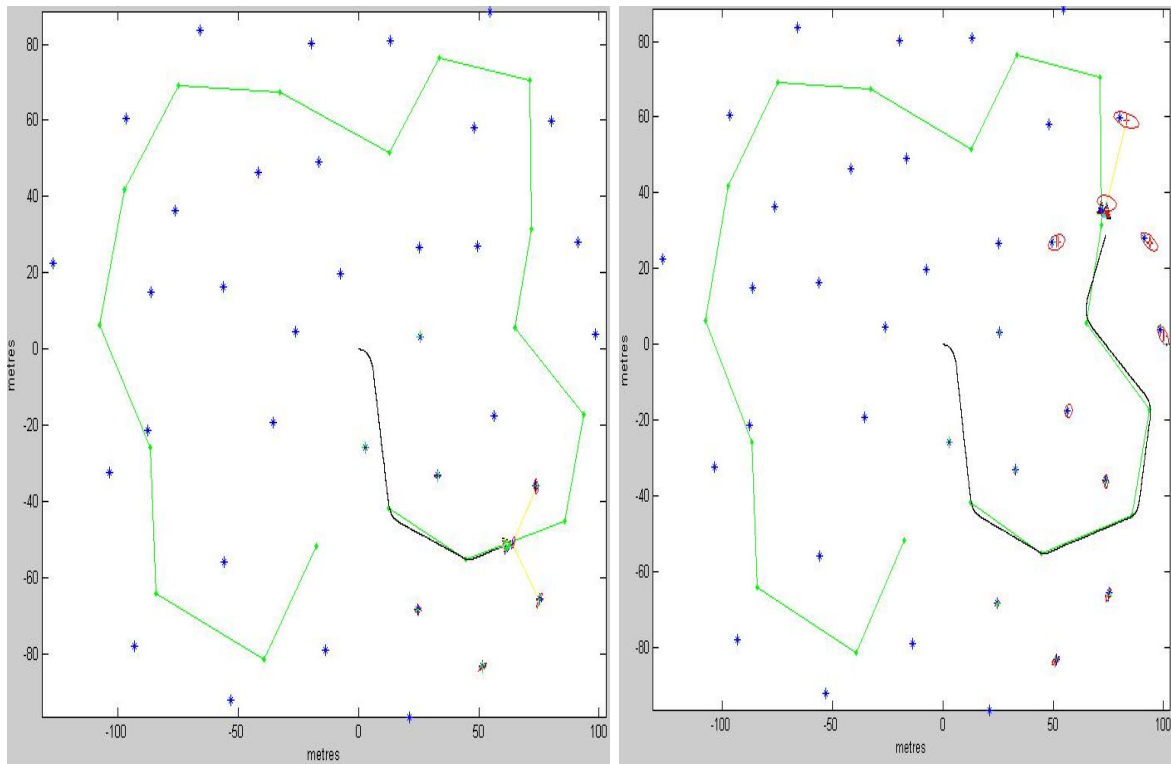
La méthode EKF-SLAM n'est applicable que dans le cas d'environnements où on peut extraire un nombre important de points pouvant caractériser l'environnement (*indoor environment*) et où les erreurs commises sur le modèle d'observation et le modèle cinématique sont acceptables, autrement cette méthode fournira un SLAM de faible qualité qui pourra facilement induire en erreur le processus décisionnel du robot mobile.

3.2.2 Simulation de la méthode UKF-SLAM

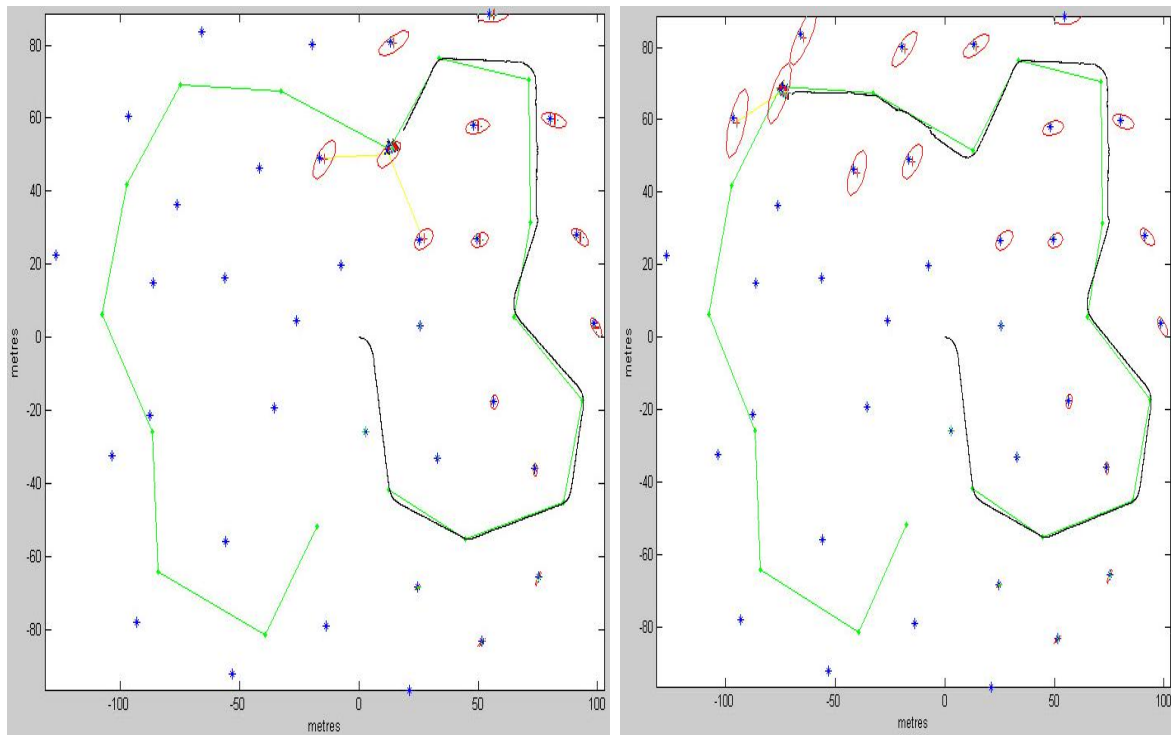
La trajectoire verte est la trajectoire réelle parcourue par le robot, la trajectoire en noir est la trajectoire estimée. Les points bleus représentent les points caractérisant l'environnement et les points rouges les points estimés. Les cercles rouges représentent l'erreur commise lors de l'estimation de la carte de l'environnement.

Résultats de la simulation

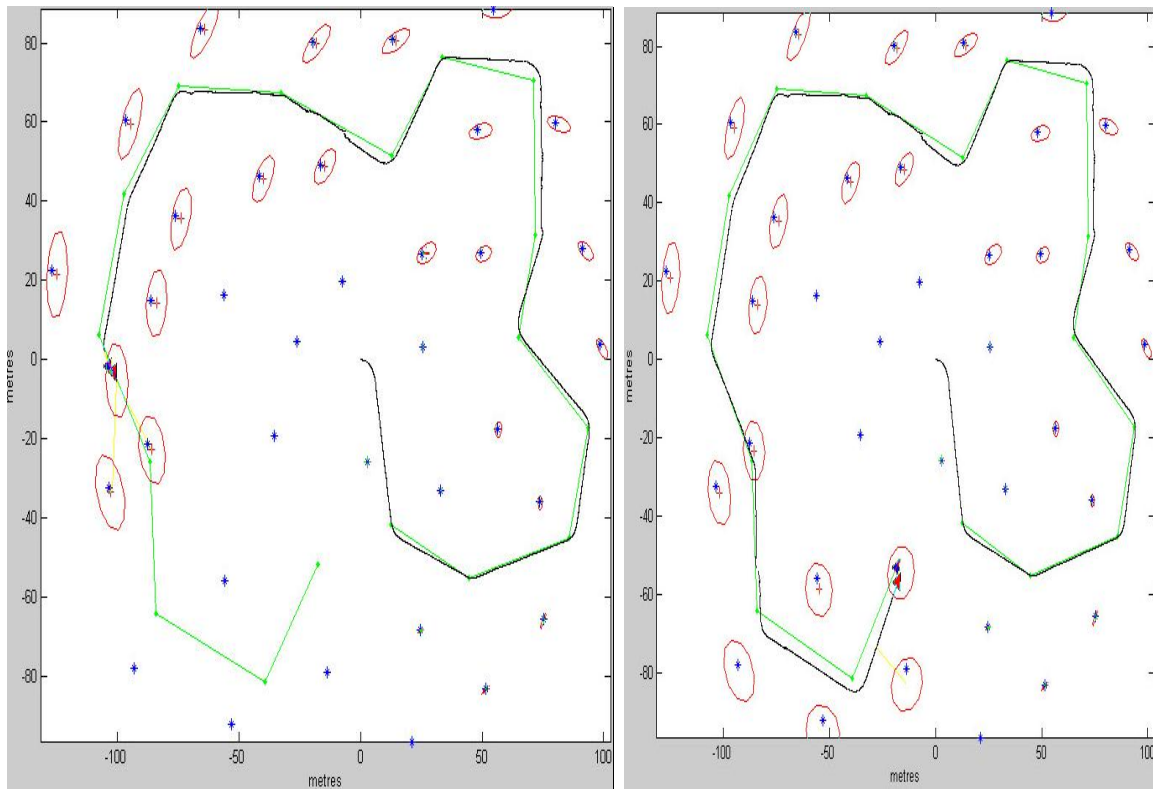
Instants t_0 et t_1 :



Instants t_2 et t_3 :



Instants t_4 et t_5 :

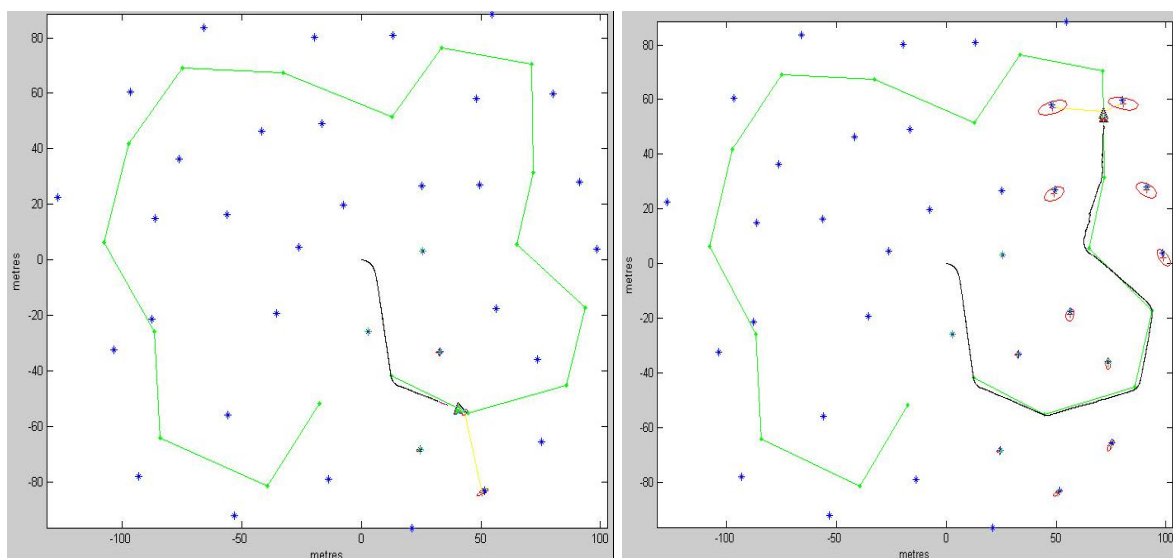


On voit que les résultats de cette simulation sont satisfaisants vu que le robot réussit à valider les points caractéristiques de l'environnement tout en estimant sa position de façon correcte.

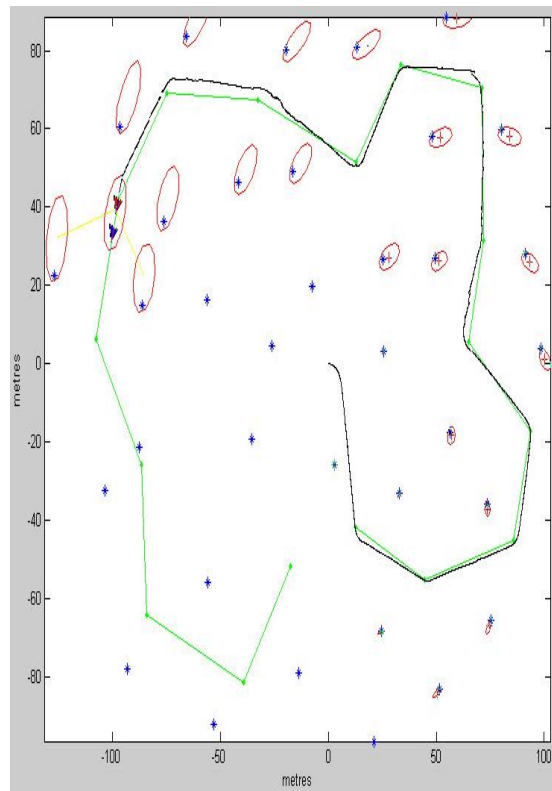
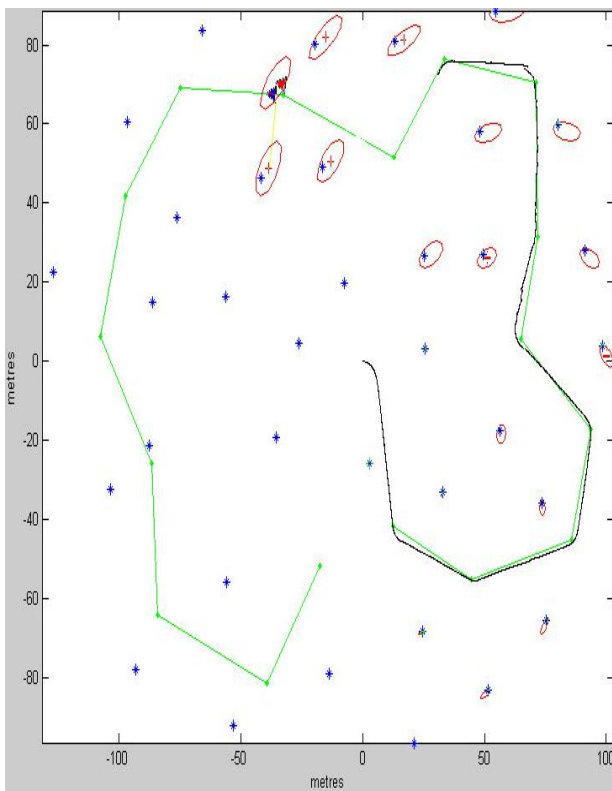
On va à présent augmenter l'erreur maximale qu'on peut avoir sur le modèle cinématique du robot à 1m/s, les résultats sont les suivants

Résultats de la simulation

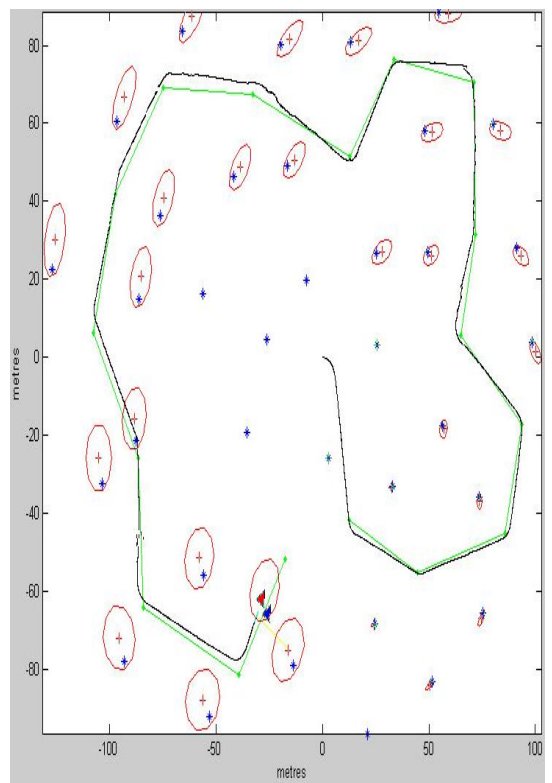
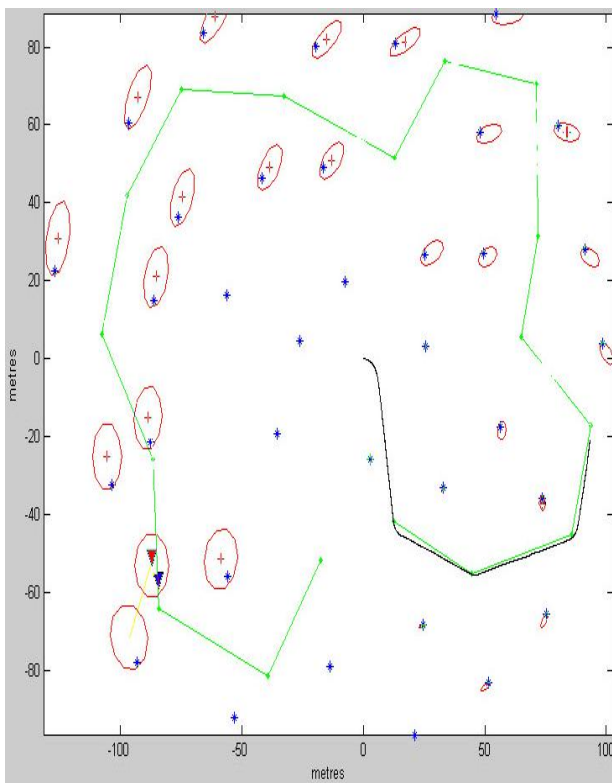
Instants t_0 et t_1 :



Instants t_2 et t_3 :



Instants t_4 et t_5 :

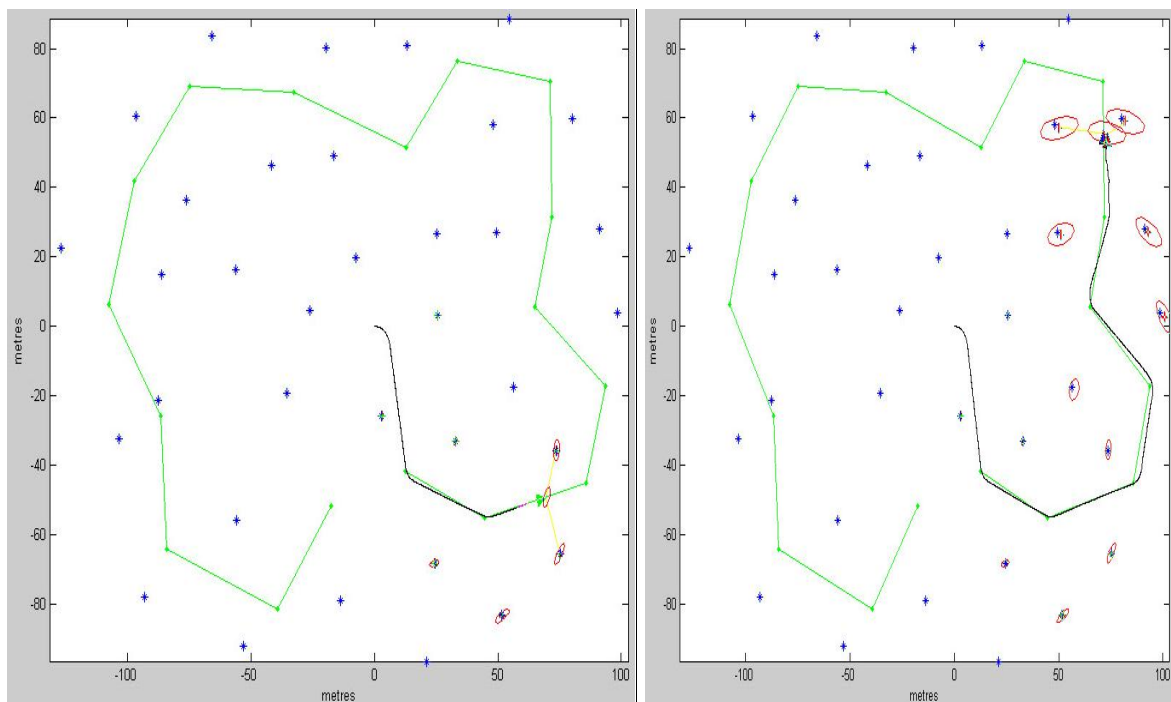


On voit bien que l'algorithme UKF-SLAM a réussi donner un résultat assez satisfaisant malgré l'importance de l'erreur commise sur le modèle cinématique. Ceci est dû au caractère du filtre de Kalman inodore qui présente une robustesse aux erreurs de modélisation.

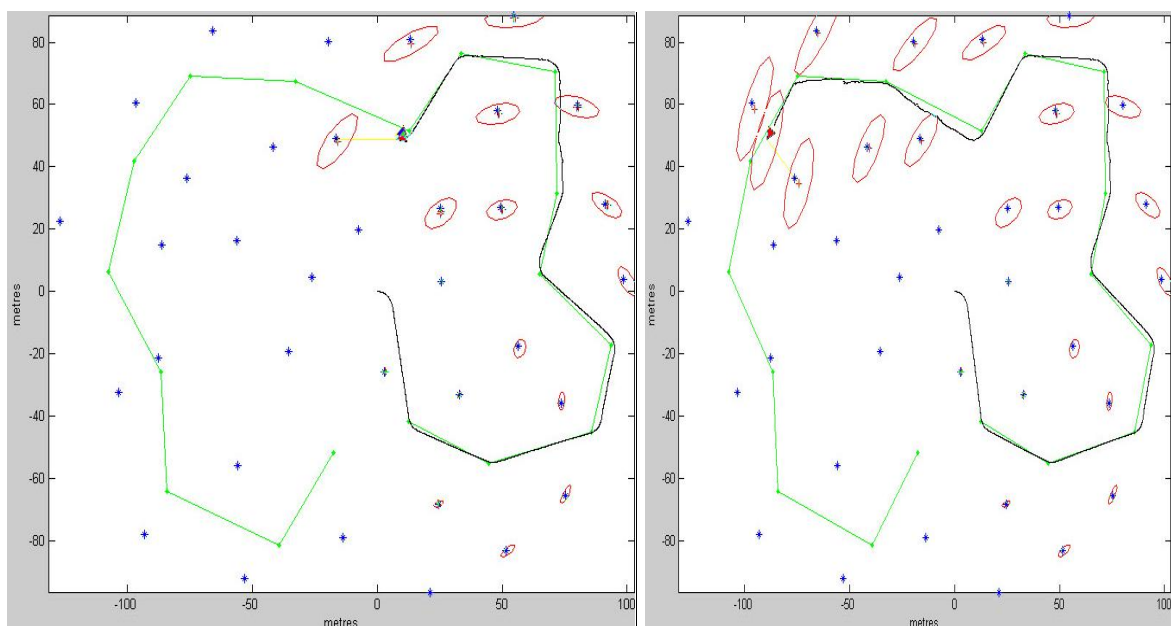
Cette dernière simulation a pour but, comme dans le cas précédent, de soumettre l'algorithme UKF-SLAM à des fortes perturbations.

Résultats de la simulation

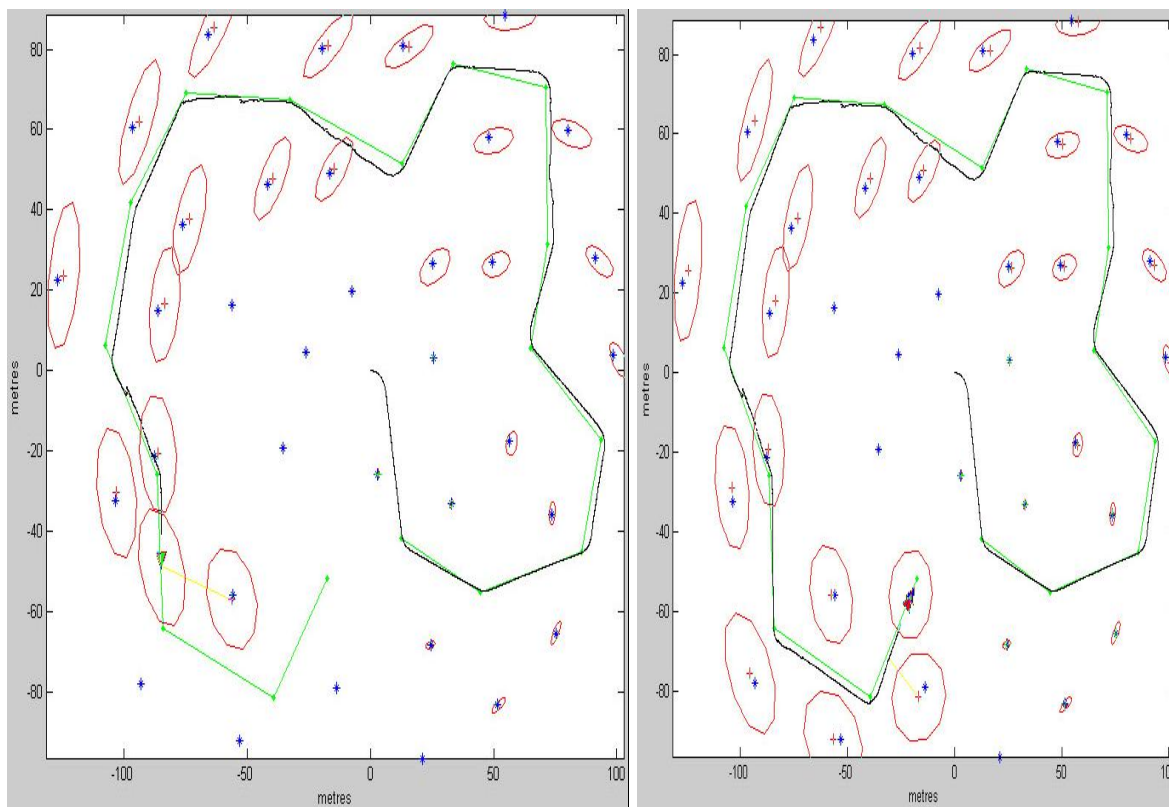
Instants t_0 et t_1 :



Instants t_2 et t_3 :



Instants t4 et t5 :



On voit bien que malgré les perturbations auxquelles le module UKF-SLAM est soumis, il réussit à donner une estimation correcte de la carte et de la position du robot

Conclusion sur la méthode UKF-SLAM

Cette méthode est très efficace vu qu'elle ne nécessite pas un nombre important de points pour caractériser l'environnement mais aussi, elle est très robuste vis-à-vis des erreurs de modélisation. Elle peut donc être utilisée dans plusieurs types d'environnements.

Conclusion sur les méthodes probabilistes basées sur un filtrage de Kalman

Les résultats de ces simulation sont satisfaisants vu qu'ils permettent une résolution satisfaisante du problème de SLAM lorsque un minimum de points pouvant caractériser l'environnement de façon correcte existent. Cependant, il est à noter que dans le cas de ces simulation, le problème d'association de données est considéré comme étant résolu ce qui est loin d'être le cas dans en pratique.

3.2.3 Simulation de la méthode ICP-SLAM

Cette méthode comme décrit dans le chapitre précédent met en correspondance des nuages de points par le biais de rotations et de translations. La figure suivante montre les résultats de simulation de l'algorithme ICP.

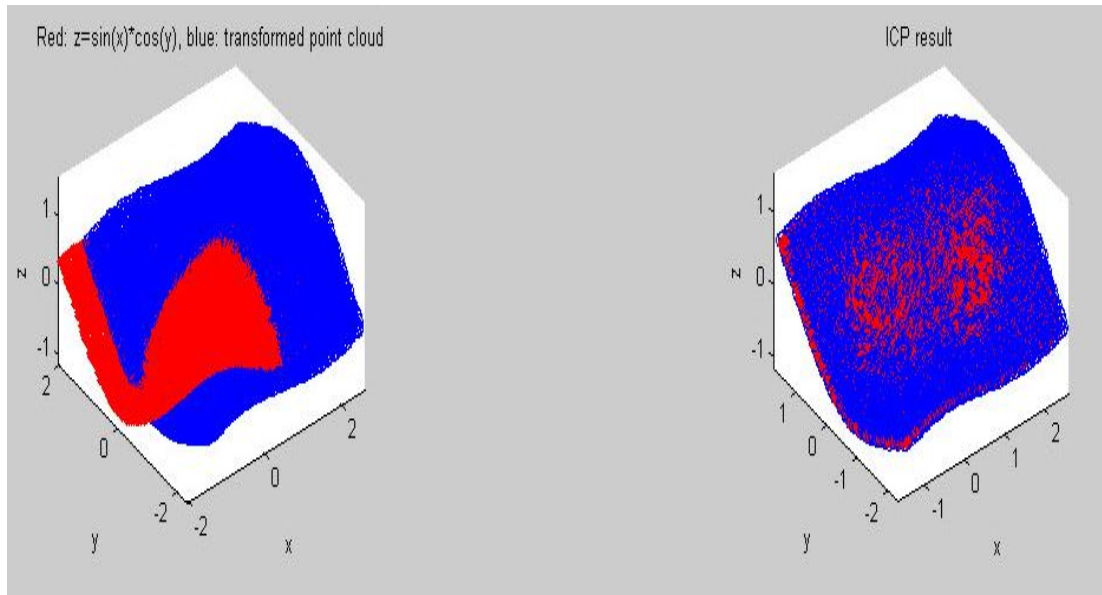


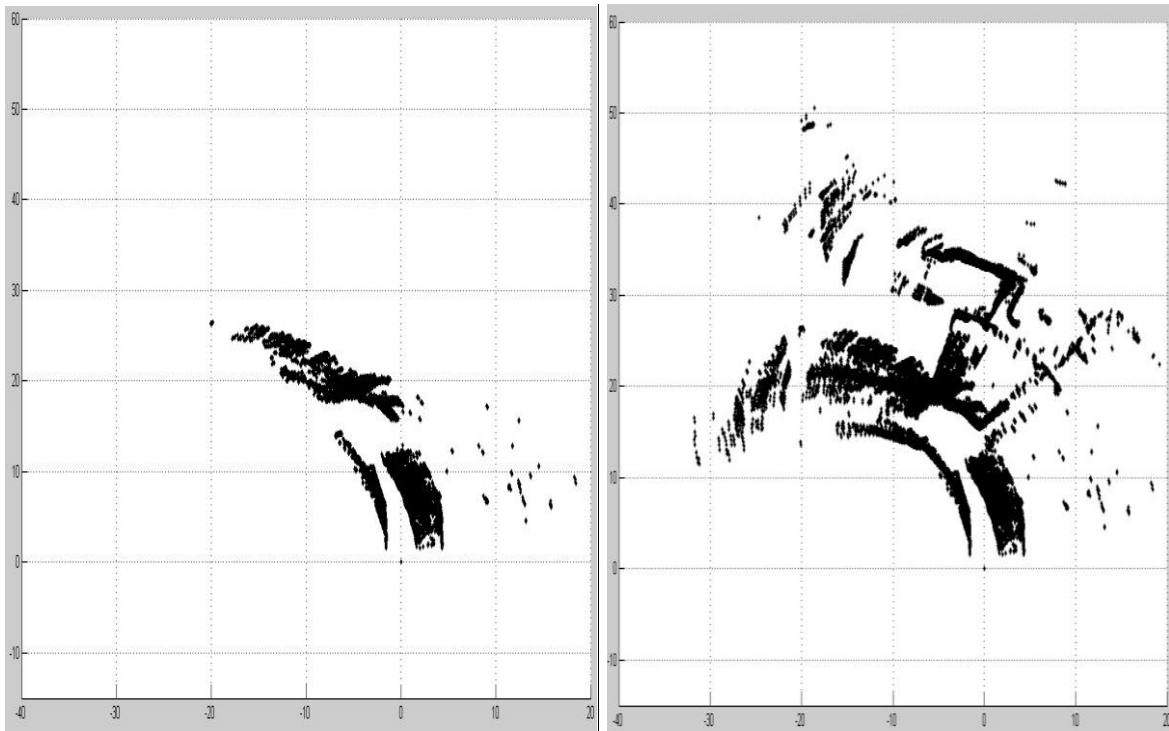
Figure 3-1 : Résultat de l'application de l'algorithme ICP

On voit bien que les deux nuages de points rouge et bleu sont mis en correspondance par l'algorithme ICP.

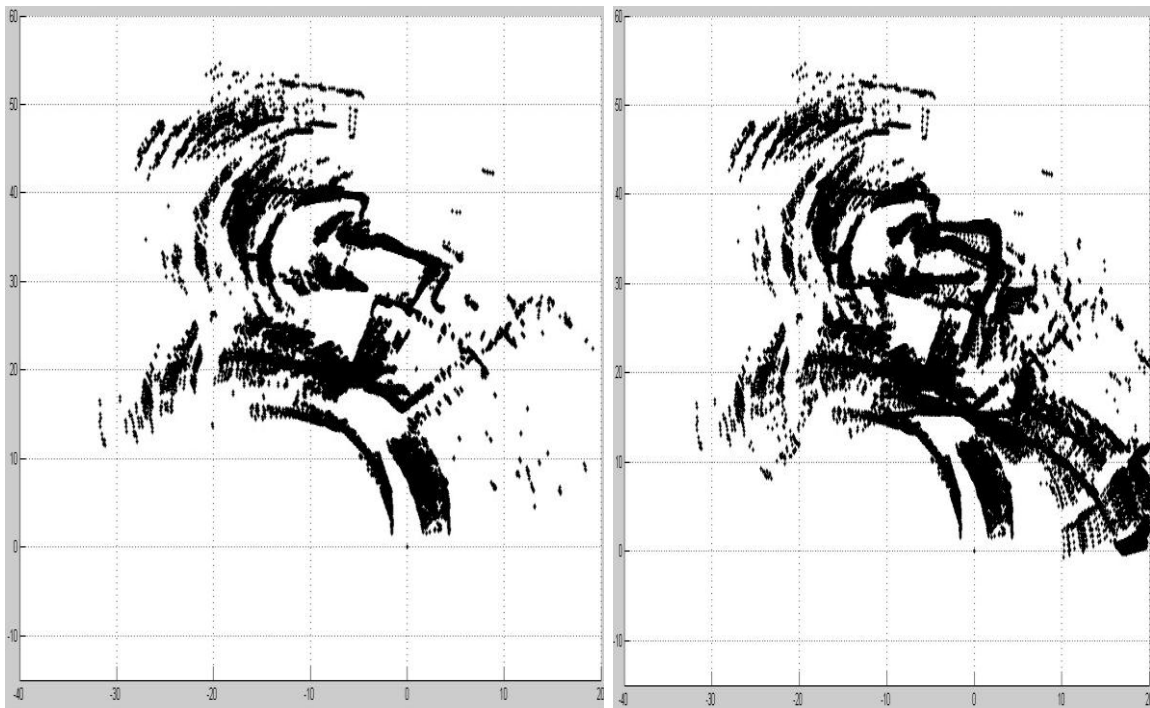
Utilisons à présent cet algorithme pour résoudre le problème de cartographie et localisation simultanée.

Les figures suivantes montrent les différents nuages de points donnés par le capteur laser du robot mobile.

Instants t_0 et t_1 :



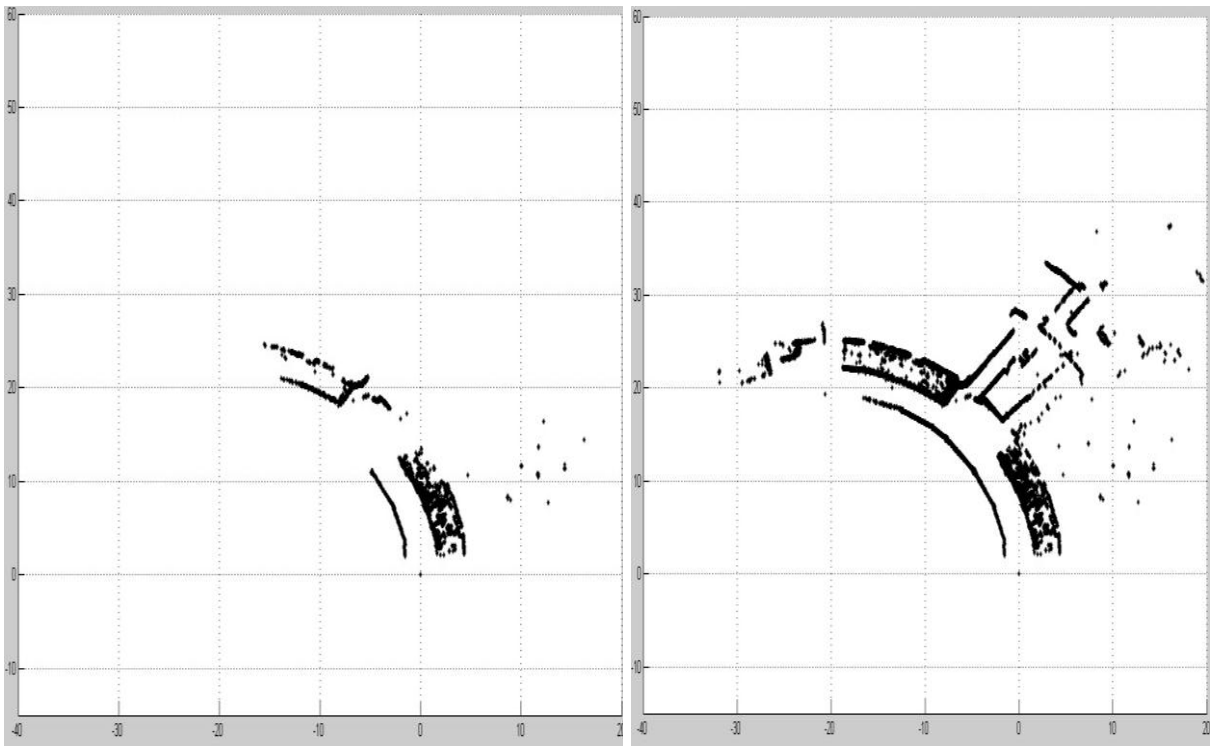
Instants t_4 et t_5 :



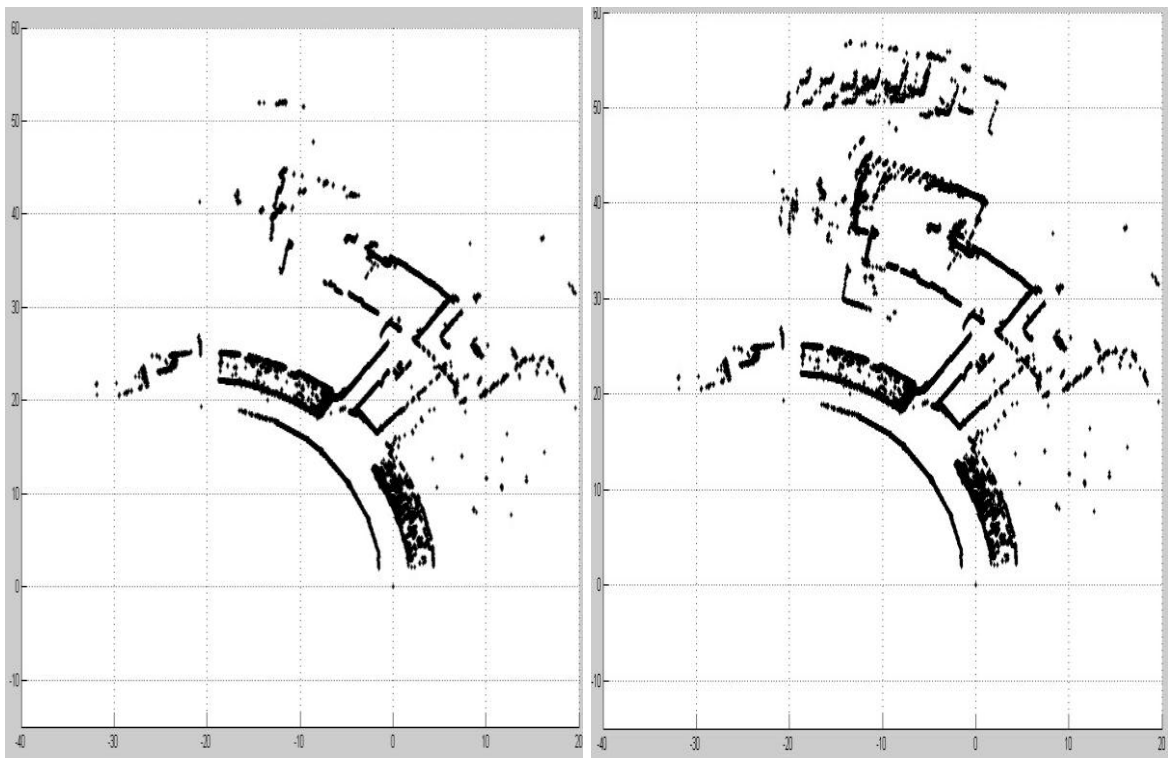
On va à présent appliquer l'algorithme ICP à ces nuages de points pour résoudre le problème de SLAM.

Résultats de la simulation

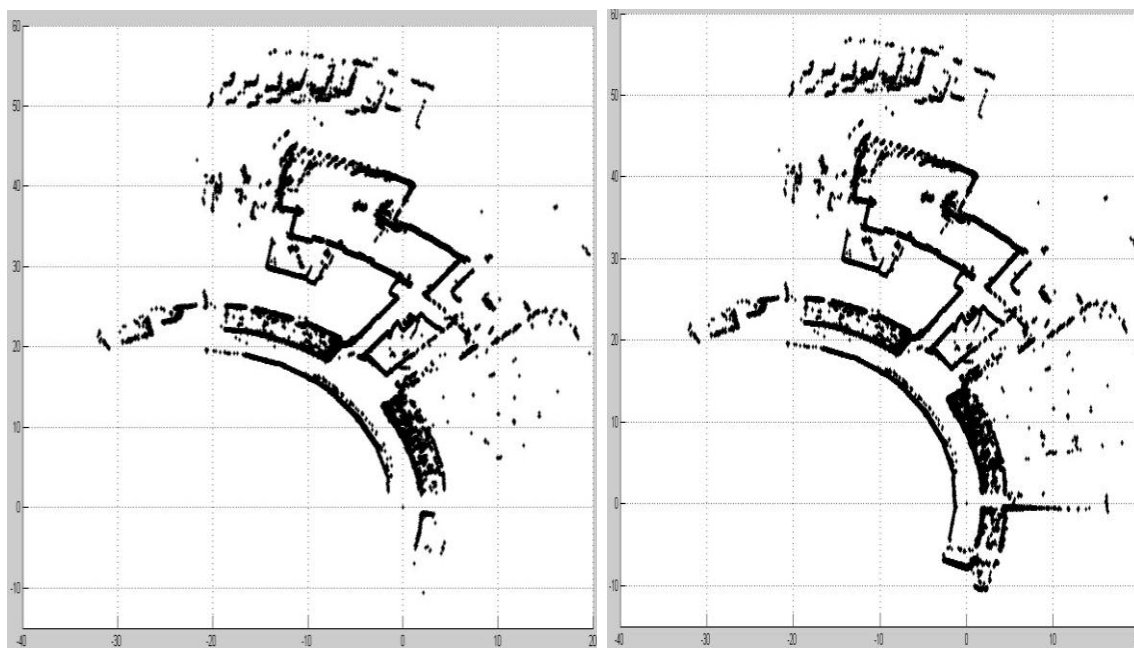
Instants t_0 et t_1 :



Instants t_2 et t_3 :



Instants t_4 et t_5 :



L'utilisation de l'algorithme ICP-SLAM permet dans le cas d'un environnement statique de donner des résultats très acceptables. En effet l'estimation de la carte reflète bien l'environnement réel dans lequel évolue le robot. Cependant l'utilisation de cette méthode requière comme hypothèse que l'environnement dans lequel évolue le robot est statique et cette hypothèse est loin d'être toujours vérifié.

3.3 Conclusion

De par les résultats des simulations, on voit bien que la résolution du problème de SLAM donne des résultats très acceptables dans le cas où l'environnement dans lequel évolue le robot est parfaitement statique.

Vu que l'hypothèse de l'environnement statique n'est pas toujours vérifiée, la nécessité de résoudre le problème de SLAM dans le cas d'environnements dynamique a motivé plusieurs recherches. L'utilisation de l'odométrie du robot pour le localiser parait comme la solution la plus naturelle. Cependant, cette solution montre très vite ses limites dans le cas d'environnements larges.

L'utilisation d'un filtrage des données appartenant aux objets dynamiques pour ensuite localiser le robot par le biais des capteurs extéroceptifs du robot est la solution la plus prometteuse vu que les données des capteurs extéroceptifs sont plus fiables.

La méthode la plus utilisée pour assurer ce filtrage est le couplage des deux modules SLAM et DATMO, ce couplage permettra d'éliminer les objets dynamiques du module de SLAM pour

assurer un résultat précis d'une part et d'une autre part permettra de fournir une perception artificielle complète au robot mobile.

Conclusion générale

Le domaine de la robotique mobile est devenu depuis quelques décennies un domaine de recherches stratégiques ou plusieurs entreprises de conception et de commercialisation de voitures investissent de grosses sommes d'argent dans le développement technologiques de leurs véhicules. Plusieurs systèmes d'aide à la conduite ont déjà été adoptés et de nos jours, le but à atteindre est de passer au stade du véhicule complètement autonome.

La première brique dans le processus de conception d'un véhicule autonome est de le rendre capable de percevoir son environnement, une tâche bien complexe qui nécessite un savoir-faire pluridisciplinaire combinant des connaissances en instrumentation, en informatique et programmation, en intelligence artificiel et en plusieurs autres domaines. Cette diversification de compétences est inévitable vu l'enjeu de ce problème de perception.

Le problème de perception artificiel dans le cadre de la robotique mobile se compose en quatre sous problèmes :

- La construction de la carte de l'environnement entourant le robot.
- La localisation du robot dans la carte construite.
- Détecter les objets mobiles présents dans l'environnement du robot.
- Prédire le mouvement de ces obstacles dynamiques.

Les deux premiers problèmes entrent dans le cadre du SLAM (Simultaneous Localization And Mapping), quant au deux derniers, ils constituent le problème de DATMO (Detection And Tracking of Moving Objects). La résolution de ces deux problèmes en simultanés permettra de donner un modèle de l'environnement du robot, avec lequel il pourra effectuer plusieurs tâches de navigation autonome (planification de trajectoire, évitement d'obstacles ...etc.).

Ce mémoire vise à compléter le travail effectué dans notre projet de fin d'études qui avait pour but la mise en œuvre d'un module de détection et de suivi d'objets mobiles (DATMO) sur la plateforme expérimentale Robucar. En effet, ce travail traite un autre module essentiel à la perception artificielle d'un véhicule autonome, à savoir, le SLAM.

On a donc commencé ce travail par situer le problème de perception artificielle, son importance, dans la chaîne PDA de tout système intelligent. On a ensuite présenter les différentes approches pouvant résoudre ce problème de SLAM pour au finale, présenter des simulations de quelqu'une de ces approches.

On a vu que le couplage des deux modules SLAM et DATMO est essentielle non seulement pour assurer une perception complète pour le robot mobile, mais aussi, c'est une façon très efficace de résoudre le problème de cartographie et de localisation simultanée. En effet, filtrer les données appartenant aux objets dynamiques pour pouvoir utiliser les données des capteurs extéroceptifs du robot pour sa localisation permet une localisation fiable contrairement au cas où on utilise des odomètres pour déduire la position du robot ce qui donne des résultats erronés lorsque l'environnement dans lequel évolue le robot est large.

Bibliographie

1. **Morette, Nicolas.** *Contribution à la Navigation de robots mobiles : approche par modèle direct et commande prédictive.* s.l. : Thèse de doctorat à l'Université d'Orléan, 2009.
2. **Qadeer, Baig.** *Multisensor Data Fusion for Detection and Tracking of Moving Objects From a Dynamic Autonomous Vehicle.* s.l. : Université de Grenoble, 2012. p. 143.
3. **DROCOURT, Cyril.** *Localisation et modélisation de l'environnement d'un robot mobile par coopération de deux capteurs omnidirectionnels.* 2002.
4. **Anna Petrovskaya, Sebastian Thrun.** *Model Based Vehicle Detection and Tracking for Autonomous Urban Driving.* 2009.
5. **A. Petrovskaya, P. Mathias, L. Oliveira, S. Luciano, R. Triebel, A. Makris, J.-D. Yoder, C. Laugier, U. Nunes et P. Bessiere,.** *Awareness of Road Scene Participants for Autonomous Driving.*
6. **Elfes.** *Using occupancy grids for mobile robot perception and navigation.* 1989.
7. **Thorpe, Chieh-Chih Wang and Chuck.** *Simultaneous Localization and Mapping with Detection and Tracking of Moving Objects.*
8. **Hugh Durrant-Whyte, Fellow and Tim Bailey.** *Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms.* 2006.
9. **Yavuz, Zeyneb Kurt-Yavuz and Sirma.** *A Comparison of EKF, UKF, FastSLAM2.0, and UKF-based FastSLAM Algorithms .* 2012.
10. **Avenel, Christophe.** *Suivi de courbes fermées déformables par filtrage stochastique.* 2007.
11. **Sebastian Thrun, Michael Montemerlo, Daphne Koller, Ben Wegbreit, Juan Nieto, and Eduardo Nebot.** *FastSLAM: An Efficient Solution to the Simultaneous Localization And Mapping Problem with Unknown Data Association.*
12. **Parr, Austin Eliazar and Ronald.** *DP-SLAM: Fast, Robust Simultaneous Localization and Mapping Without Predetermined Landmarks.*
13. **Javier Gonzalez, Anthony Stentz and Anibal Ollero.** *A Mobile Robot Iconic Position Estimator using a Radial Laser Scanner.* 1992.
14. **Shaffer, Gary K.** *Two-Dimensional Mapping of Expansive Unknown Areas.* 1995.
15. **Cox, Ingemar J.** *Blanche- An Experiment in Guidance and Navigation of an Autonomous Robot Vehicle.* s.l. : IEEE, 1991.

16. **Quigley, Morgan, et al., et al.** *ROS: an open-source Robot Operating System*. 2009.
17. **Hugh Durrant-Whyte, Fellow, IEEE, and Tim Bailey.** *Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms*. 2006.
18. **Fuerstenberg, Kay Ch., et al., et al.** Multilayer laserscanner for robust object tracking and classification in urban. *In 9th World Congress on Intelligent Transport Systems*. 2002, p. 8.