



HAL
open science

Localisation robuste en milieu industriel complexe

Pierre Merriaux, Yohan Dupuis, Rémi Boutteau, Pascal Vasseur, Xavier Savatier

► **To cite this version:**

Pierre Merriaux, Yohan Dupuis, Rémi Boutteau, Pascal Vasseur, Xavier Savatier. Localisation robuste en milieu industriel complexe. Colloque GRETSI, Sep 2015, Lyon, France. hal-01875375

HAL Id: hal-01875375

<https://hal.science/hal-01875375>

Submitted on 17 Sep 2018

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.

Localisation robuste en milieu industriel complexe

Pierre MERRIAUX¹, Yohan DUPUIS², Rémi BOUTTEAU¹, Pascal VASSEUR³ Xavier SAVATIER¹

¹Laboratoire IRSEEM/ESIGLEC
76800 Saint-Etienne-du-Rouvray, France

²Laboratoire CEREMA,
76120 Le Grand Quevilly, France

³Laboratoire LITIS,
76821 Saint-Etienne-du-Rouvray, France

prénom.nom@esigelec.fr, yohan.dupuis@cerema.fr
pascal.vasseur@univ-rouen.fr

Résumé – Dans ce papier nous proposons une solution de localisation en milieu industriel complexe basée lidar et filtrage particulière. Notre méthode de localisation en 6 degrés de liberté (ddl) utilise une nouvelle fonction de vraisemblance efficace en temps de calcul grâce à un champ de vraisemblance 3D pré-calculé. Au final, nous obtenons une localisation 6 ddl temps-réel, avec des erreurs de positionnement et d'angle inférieures à 2cm et 1° dans un environnement de **350m³**.

Abstract – In this paper, we propose a LIDAR-based robot localization method in a complex industrial environment. Localization is achieved in 6 Degrees of freedom (DoF) thanks to the particle filter framework. A new time-efficient likelihood function, based on a pre-calculated 3D likelihood field, is introduced. Experiments are carried out on both simulated and real environments. Six DoF real-time localization is achieved with a spatial and angular errors of less than 2cm and 1° respectively in an real environment of **350m³**.

1 INTRODUCTION

La localisation en milieu industriel peut être aujourd'hui résolue par différentes techniques. Elles nécessitent très souvent l'équipement du site par des lignes [?] ou des balises réfléchissantes [?]. De plus l'ensemble de ces méthodes apporte une solution pour une localisation dans un milieu ordonné, essentiellement composé d'un ensemble d'allées et d'îlots de rangement. Dans le cadre d'une localisation en milieu complexe, par exemple une industrie chimique ou pétrolière, les unités sont composées d'allées étroites et d'un enchevêtrement de tuyaux, pompes, chaudières . . . s'étalant sur plusieurs niveaux. La structure irrégulière des allées, la nécessité de franchir des obstacles et des escaliers pour changer de niveau imposent une localisation complète dans les six degrés de liberté. La complexité et l'étendue souvent très importantes des environnements industriels rendent difficile le déploiement de solutions de localisation nécessitant l'instrumentation du site.

La localisation [?] s'approche d'une solution plus réaliste pour l'application envisagée. Ils utilisent l'information 3D d'une Kinect afin de calculer la vraisemblance avec une simulation 3D. Bien qu'obtenant des résultats intéressants, le fait de calculer l'équivalent d'un lancer de rayon pour chaque point 3D de chaque hypothèse de position, limite considérablement les performances. De plus la solution ayant été développée pour la localisation en intérieur de type bureau, son modèle 3D est

limité à l'extraction de plans de surfaces importantes, ce qui ne peut convenir au type d'environnement complexe ciblé par nos travaux.

Dans ce papier nous proposons une solution de localisation 6 degrés de liberté (ddl), basée sur la généralisation à trois dimensions de la méthode des champs de vraisemblance appliquée sur des données LIDAR. Notre méthode fonctionne en environnement complexe, et la rapidité de calcul permet d'évaluer un grand nombre d'hypothèses à la fois. Au final, nous obtenons une localisation temps réel, 6 ddl, et robuste aux variations d'environnement.

2 METHODOLOGIE

2.1 Contexte

Les sites industriels visés comportent des géométries complexes, de nombreuses symétries et similarités (figure ??). Les contraintes opérationnelles, comme le fonctionnement jour et nuit, en toutes conditions météorologiques et le niveau de sûreté de fonctionnement exigé, nous ont poussé à écarter les solutions basées vision pour nous orienter vers les LIDARS.

2.2 Théorie de la localisation

Le but de la localisation est de déterminer le vecteur d'état \mathbf{X}_t le plus probable, étant donné une carte \mathcal{M} de l'environne-

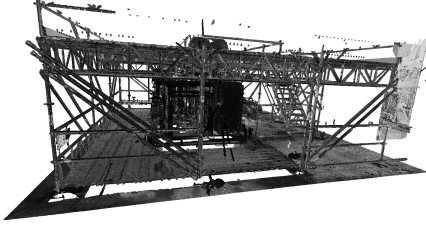


FIGURE 1 – Nuage de points 3D issu de la numérisation d'un site industriel complexe

ment connue à priori, et un capteur nous fournissant des mesures \mathbf{Z}_t sur cet environnement. Le vecteur d'état \mathbf{X}_t est défini de la manière suivante :

$$\mathbf{X}_t = [x \ y \ z \ \psi \ \theta \ \varphi]^T \quad (1)$$

Où :

- x, y, z : Position en mètre (m)
- ψ, θ, φ : Orientation en degré (°)

Pour les applications de localisation, il existe de nombreux filtres dans la littérature. Dans ces travaux nous avons utilisé le filtre particulaire car il n'est pas limité à une distribution unimodale de probabilité du vecteur d'état [?].

Nous avons donc implémenté un filtre particulaire standard [?], pour estimer le vecteur d'état $\hat{\mathbf{X}}_t$. Son rôle est d'émettre des hypothèses de $\hat{\mathbf{X}}_t$, appelées particules, et de les sélectionner à l'aide de la fonction de vraisemblance pour converger vers une solution. $\hat{\mathbf{X}}_t$ est le résultat du barycentre des particules.

Dans ce processus la fonction de vraisemblance joue un rôle essentiel. Elle va évaluer les différentes hypothèses pour sélectionner les plus probables ou autrement dit, la probabilité d'obtenir une mesure \mathbf{Z} , pour un état \mathbf{X} , en connaissant la carte \mathcal{M} a priori : $P(\mathbf{Z}|\mathbf{X}, \mathcal{M})$.

La section ?? présente quelques méthodes d'évaluation de la fonction de vraisemblance proposée. Puis, nous introduisons un espace pré-calculé de probabilités d'impact LIDAR, appelé "champ de vraisemblance", et proposons une variante 3D de cette méthode.

2.3 Champ de vraisemblance 3D

Il y a plusieurs façons d'interpréter les informations d'un LIDAR, par exemple pour la localisation d'un véhicule, [?] utilise l'information 3D pour projeter les impacts sur le plan constituant la route, ce qui lui permet de faire une comparaison basée sur la réflectivité infrarouge du laser et une carte pré-établie. En robotique mobile extérieure en environnement non structuré, le concept de "Multi-Level Surface Maps" [?] est couramment utilisé. Cela consiste à utiliser la hauteur de l'environnement proche, et le comparer également à une carte pré-établie.

Une méthode similaire à [?] ne peut convenir à notre application pour deux raisons : notre environnement étant composé d'une multitude de petits tuyaux, il faut énormément de facettes pour le représenter correctement, et la fonction de vraisemblance obtenue est très sensible à une variation de \mathbf{X} .

Pour palier ces deux inconvénients nous proposons d'étendre l'utilisation d'un "champ de vraisemblance" [?][?] utilisé dans le plan à un modèle 3D. L'environnement est représenté par une grille 3D, et pour chaque objet constituant notre scène, nous calculons la probabilité d'impact LIDAR avec une loi normale :

$$p_{hit}(\mathbf{z}|m, O_{\mathcal{M}}^k) = \frac{1}{\sigma_{map}\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{d}{\sigma_{map}}\right)^2} \quad (2)$$

Où :

- m : cellule de vraisemblance
- d : Distance euclidienne entre la cellule m et le k^{th} obstacle $O_{\mathcal{M}}^k$ dans \mathcal{M}
- σ_{map} : incertitude de la carte

Chaque case de la grille 3D de la carte va constituer le champ de vraisemblance de la manière suivante :

$$p_{hit}(m) = \max_{0 \leq k \leq n_{obstacle}} p_{hit}(\mathbf{z}|m, O_{\mathcal{M}}^k) \quad (3)$$

Où :

- $n_{obstacle}$: nombre d'obstacles dans l'environnement

Le champ de vraisemblance $\mathcal{L}(\mathcal{M})$, est une grille 3D $p_{hit}(m)$ de cellules m qui discrétisent \mathcal{M} .

La résolution angulaire du LIDAR détermine le nombre d'impacts (nb) possibles. Seulement un sous ensemble \mathbb{L} de ces impacts seront mesurés. En effet, l'objet peut être trop loin, la surface trop absorbante ou l'angle d'incidence trop faible.

Grâce à l'équation ??, la détermination de $p(\mathbf{Z}|\mathbf{X}, \mathcal{M})$ est très aisée. Il faut commencer par projeter les points LIDAR $\mathbf{P}_{\mathcal{L}}$ dans le repère robot \mathcal{R} puis dans le repère de la carte \mathcal{M} (figure ??) :

$$\mathbf{P}_{\mathcal{M}} = \begin{bmatrix} \mathbf{R}_{\mathcal{R} \rightarrow \mathcal{M}} & \mathbf{t}_{\mathcal{R} \rightarrow \mathcal{M}} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{\mathcal{L} \rightarrow \mathcal{R}} & \mathbf{t}_{\mathcal{L} \rightarrow \mathcal{R}} \\ 0 & 1 \end{bmatrix} \cdot \mathbf{P}_{\mathcal{L}} \quad (4)$$

Puis calculer le score $\mathbb{P}(\mathbf{Z}|\mathbf{X}, \mathcal{M})$ directement en relisant le champ de vraisemblance $\mathcal{L}(\mathcal{M})$:

$$p(\mathbf{Z}|\mathbf{X}, \mathcal{M}) = \frac{(\sum_{i \in \mathbb{L}} p_{hit}(m_i | p_{i, \mathcal{M}} \in m_i))^2}{n} \quad (5)$$

Où :

- $p_{i, \mathcal{M}}$: i^{th} impact du LIDAR de la mesure $\mathbf{P}_{\mathcal{M}}$.

Les impacts LIDAR étant considérés comme indépendants, l'équation ?? ne devrait pas être une somme. Mais un produit dépasse rapidement les capacités de codage même sur un flotant 64 bits. Différentes types de somme sont présentés dans la thèse [?].

3 Résultats expérimentaux

Cette section présente les résultats obtenus dans deux environnements.

Le développement rapide de LIDARs multi-nappes dans l'industrie nous a poussé à évaluer l'apport d'un LIDAR multi-nappes par rapport au classique mono-nappe. Les résultats présentés dans cette section sont donc basés sur les caractéristiques des LIDARs suivants :

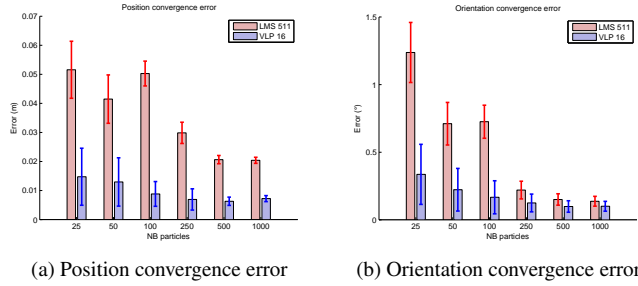


FIGURE 2 – Moyenne et écart type de la précision de la convergence en fonction du nombre de particules.

- le cas du mono-nappe (*SL* : single layer) sur le Sick LMS511 ¹
- le cas du multi-nappes (*ML* : multi layers) sur le Velodyne VLP16 ²

3.1 Étude de la convergence

Afin de tester la convergence de notre solution nous nous mettons dans la situation du robot "perdu" : nous initialisons le filtre particulaire à une pose X_{init} comportant une erreur (ϵ_{pos}) de la position réelle X_{truth} ($X_{init} = X_{truth} + \epsilon_{pos}$). Dans ce cas il n'y a pas d'information pour l'étape de *motion update*, mais juste un bruit additif pour assurer la convergence. Les poses des particules sont initialisées avec une variance égale à l'erreur de position, $\sigma_{init} = abs(\epsilon_{pos})$, $\sigma_{prop} = 0$ et $\sigma_{add} = \sigma_{init}/10$.

Comme métrique nous utilisons, pour la position la distance euclidienne et pour l'orientation nous calculons l'angle solide 3D par le biais de la similarité en cosinus : $\Omega = \arccos \frac{X_t \cdot \hat{X}_t}{\|X_t\| \cdot \|\hat{X}_t\|}$

La figure ?? présente la précision de la convergence en position et en orientation en fonction du nombre de particules, pour une position initiale erronée de $\epsilon_{pos} = [0.2 \ -0.2 \ 0.05 \ 2^\circ \ -2^\circ \ 5^\circ]$. La convergence est calculée sur 200 itérations de *measurement update*, et le résultat est déterminé avec la moyenne et l'écart type des 50 dernières itérations. Ces tests sont répétés 8 fois afin de s'affranchir de la variabilité des résultats quand le nombre de particules est faible. Ce sont les moyennes des 8 essais qui sont présentées. Le bruit LIDAR est issu d'un nouveau tirage pour chaque itération, et sa variance fixée à $\sigma_{lidar} = 0.01m$. Le nombre de particules idéal se situe entre 250 et 500.

3.2 Localisation

Pour tester le processus de localisation complet, nous avons généré avec le robot une trajectoire constituée de 160 poses LIDAR ainsi que l'odométrie correspondante (figure ??) dans l'environnement de la figure ?. Sa longueur est de 22.1m. Le bruit LIDAR est de $\sigma_{lidar} = 0.01m$. L'odométrie est bruitée proportionnellement aux déplacements effectués, 10% en linéaire

1. http://www.sick.com/instruments/EN/home/products/flow_sensors/Pages/BulkscanLMS511.aspx
 2. <http://velodynelidar.com/lidar/hdlproducts/vlp16.aspx>

TABLE 1 – Performance globale de la localisation sur une trajectoire de 22.1m, 500 particules, $\sigma_{lidar} = 1cm$.

	Position error (cm)		Orientation error (°)		computation time Intel i2640M-2.8GHz (ms)
	mean	std	mean	std	
SL	2.23	1.45	0.45	0.81	0.9 (381 LIDAR impacts)
ML	1.57	1.20	0.31	0.44	2.2 (5760 LIDAR impacts)

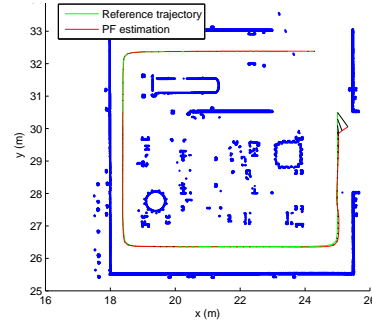


FIGURE 3 – Résultats de la localisation dans un environnement industriel pour un LIDAR mono-nappe, avec 500 particules. Référence (*vert*), trajectoire estimée par le filtre particulaire (*rouge*), coupe de l'environnement (*bleu*).

et 20% en rotation. Les résultats sont obtenus avec 500 particules. La position initiale est altérée de $\epsilon_{pos} = [0.5 \ -0.5 \ 0.05 \ 1^\circ \ -1^\circ \ 5^\circ]$.

Malgré une erreur initiale de position de 71cm et d'orientation de 5.1° , l'erreur de position moyenne est d'ordre centimétrique et l'erreur d'orientation moyenne est inférieure au degré comme le montre la table ??.

Les temps de calcul de la fonction de vraisemblance obtenus avec Matlab sont compatibles avec le temps-réel comme le montre la table ??.

3.3 Résultats sur données réelles

Nous avons réalisé les expérimentations sur un robot jaguar ³ dans le laboratoire de navigation autonome de l'Irseem (figure ??). Cette pièce de $150m^2$ est équipée de 20 caméras Vicon T40S ⁴ nous servant de vérité terrain. L'odométrie et le LIDAR embarqués sur le robot sont échantillonnés à 25Hz, la vérité terrain à 100Hz. L'ensemble des données sont enregistrées avec le middleware RTMaps ⁵ qui permet un datage et un rejeu synchronisé.

L'environnement a été numérisé avec un Leica ScanStation C10 ⁶, puis le champ de vraisemblance a été calculé à partir du nuage de points obtenu (section ??).

Plusieurs séquences ont été exécutées avec un LIDAR SL LMS511. Comme le montre la figure ??, le centre optique du LIDAR n'est confondu avec le centre mécanique du robot. Une transformation doit être trouvée pour projeter les impacts LIDAR dans le référentiel du robot. Nous avons choisi d'igno-

3. http://jaguar.drrobot.com/specification_V4.asp
 4. <http://www.vicon.com/System/TSeries>
 5. <http://intempora.com/>
 6. http://www.leica-geosystems.fr/fr/Leica-ScanStation-C10_79411.htm

TABLE 2 – Résultats des expérimentations réelles sur 150m² dans le laboratoire de navigation autonome de l’Irseem, 500 particules

Sequence and length (m)	sampling type	Position error (cm)			Orientation error (°)	
		initial	mean	std	mean	std
A : 14.9	25Hz	57	1.06	0.54	0.963	0.567
B : 17.1		82	1.92	1.28	0.804	0.389
C : 8.7		28	1.78	0.96	0.890	0.435
D : 14.9	0.13m	57	2.79	1.85	0.959	1.699

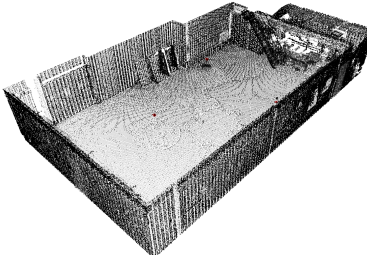


FIGURE 4 – Nuage de points du laboratoire de navigation autonome de l’Irseem

rer cette transformation, et de modéliser cette erreur par l’ajout d’un bruit additif σ_a dans notre modèle odométrique.

L’algorithme détermine la pose avec 500 particules, et est initialisé avec une erreur par rapport à la position réelle du robot. Les résultats sont présentés dans la table ???. Dans le but de comparer les performances avec l’environnement utilisé en simulation, nous avons échantillonné les données selon deux méthodes.

La première approche, utilise toutes les données des capteurs (séquences A-C). La localisation est calculée à 25Hz, ce qui donne des pas entre 3 et 7mm. Les performances sont du même ordre de grandeur que les simulations dans le premier environnement.

Pour la seconde approche, nous intégrons les données odométriques de la séquence A sur 13.8cm de manière à obtenir la même densité de mesures qu’en simulation, il en résulte la séquence D. L’erreur de position reste toujours très proche des résultats de simulation. L’erreur en rotation est un peu plus importante, ce qui peut s’expliquer par notre manque de calibration LIDAR/Robot. Notre méthode est robuste aux deux manières d’échantillonner les données

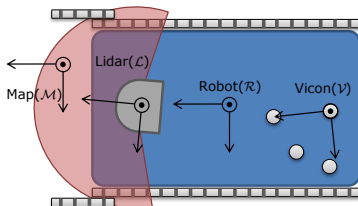


FIGURE 5 – Référentiel du robot Jaguar utilisé pour les expérimentations

4 CONCLUSION

Dans ce papier nous proposons une localisation en 6 ddl robuste et rapide basée sur l’extension des champs de vraisemblance en 3D. Le champ de vraisemblance est stocké dans une structure de données optimisée permettant de calculer une localisation en temps réel. Les performances ont été validées en simulation puis en réel dans deux environnements réels complexes. Dans un travail futur, nous améliorerons la calibration entre le LIDAR et le repère du robot, et nous testerons la méthode sur d’autres environnements complexes.

REMERCIEMENTS

Cette étude a été financée dans le cadre de notre participation au challenge international de robotique ARGOS. Le challenge ARGOS est organisé par TOTAL en partenariat avec l’Agence Nationale de la Recherche (ANR). Le nuage de points représenté figure ?? est issu de données appartenant à TOTAL.

Références

- [1] M. A. Olivares-Mendez, I. Mellado, P. Campoy, I. Mondragon, and C. Martinez, “A visual agv-urban car using fuzzy control,” in *Automation, Robotics and Applications (ICARA), 2011 5th International Conference on*. IEEE, 2011, pp. 145–150.
- [2] D. Ronzoni, R. Olmi, C. Secchi, and C. Fantuzzi, “Agv global localization using indistinguishable artificial landmarks,” in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 287–292.
- [3] M. F. Fallon, H. Johannsson, and J. J. Leonard, “Efficient scene simulation for robust monte carlo localization using an rgb-d camera,” in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 1663–1670.
- [4] S. Thrun, W. Burgard, D. Fox *et al.*, *Probabilistic robotics*. MIT press Cambridge, 2005, vol. 1.
- [5] J. Levinson and S. Thrun, “Robust vehicle localization in urban environments using probabilistic maps,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4372–4378.
- [6] P. Pfaff, R. Kümmerle, D. Joho, C. Stachniss, R. Triebel, and W. Burgard, “Navigation in combined outdoor and indoor environments using multi-level surface maps,” *WS on Safe Navigation in Open and Dynamic Environments, IROS*, vol. 7, 2007.
- [7] O. El Hamzaoui, “Localisation et cartographie simultanées pour un robot mobile équipé d’un laser à balayage : Core-slam,” Ph.D. dissertation, Ecole Nationale Supérieure des Mines de Paris, 2012.