

Construction d'un robot sous-marin pour le concours SAUC-E

Rapport d'avancement T0+6 mois (convention MRIS 2010-2011)

J. Sliwka, F. Le Bars, L. Jaulin

ENSTA Bretagne

22 Avril 2011

Descriptif de l'étude

Dans le cadre de cette étude, nous cherchons à réaliser des plateformes robotiques sous-marines dans le but de valider des concepts/algorithmes divers. Les thématiques liées à ce projet sont variées : réalisation d'une mécanique simple et robuste, télé-opération, autonomie, localisation et cartographie, détection et reconnaissance d'objets, collaboration entre robots... Pour motiver les personnes travaillant sur le projet et pour se donner des échéances, comme tous les ans, nous participons au concours SAUC-E (Student Autonomous Underwater Challenge - Europe), qui a eu lieu cette année du 4 Juillet au 10 Juillet 2011 à La Spezia en Italie.



Figure 1 : Image de la zone de compétition à La Spezia

Epreuves du concours SAUC'E 2011

Le sous-marin autonome doit effectuer une suite de missions décrite ci-dessous en totale autonomie. La télé opération à distance, l'utilisation de GPS ou toute communication avec le robot sont donc interdites durant toute tentative de parcours de ces missions.

Avant le début du parcours de missions, les équipes seront amenées à attacher une balise acoustique sur leur sous-marin ce qui permettra au jury de suivre sa position. Le dispositif fonctionne avec une fréquence de 37KHz.

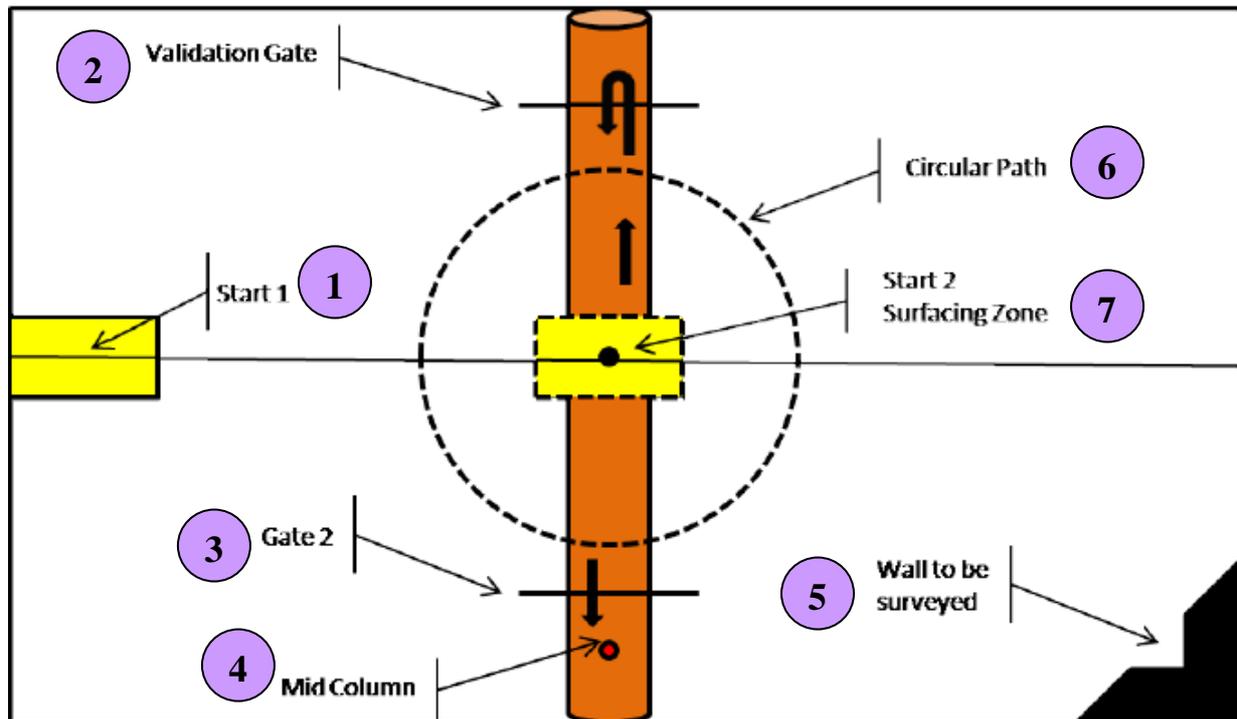


Figure 2 : Les missions du concours 2011

Mission n°1

Faire un départ d'une zone de départ spécifiée à l'avance; Les équipes ont le choix de lancer leur sous-marin de la zone Start1 ou la zone Start2. Le choix de la zone Start1 offrira un bonus de 100 points à l'équipe. Si toutefois la zone Start2 est choisie, le robot de surface du NURC y amènera le sous-marin.

Une fois submergé, le sous-marin doit passer par la porte de validation (Validation Gate) sans y toucher. La porte de validation est construite avec deux bouées orange attachées à des cordes séparées de 4m. Des LED's lumineuses seront placées sur les cordes pour faciliter leur repérage.

La mission 1° doit être effectuée au début du parcours de missions. L'échec de cette première mission interdit la suite du parcours des missions.

Mission n°2

Faire l'inspection d'une structure sous-marine. La structure sera construite avec des morceaux de tubes PVC de couleur jaune de 50cm de diamètre et de 1.5m de long. La structure obtenue dite « pipeline » ne sera pas nécessairement droite. La structure sera mise au fond et sera modifiée régulièrement. Le but est que le sous-marin suive le pipeline tout en maintenant une distance de 0.5m.

Mission n°3

Libérer une bouée. Une bouée ronde de couleur orange sera maintenue sous l'eau par un fil de pêche de 1mm de diamètre tendu avec une tension de 5Kg. Le but de la mission sera de localiser la bouée, la contourner i.e. effectuer une manœuvre d'évitement d'obstacle, retourner au point d'approche initial (en ce point 50% de la mission sera considérée comme accompli) et enfin approcher la bouée dans le but de découper le fil de pêche qui la retient.

Mission n°4

Effectuer un suivi de mur. Ce mur sera approximativement à 10m de la bouée à libérer. L'objectif de la mission est de maintenir une distance de plus de 2m du mur pendant le parcours. Le mur ne sera pas nécessairement droit et le sous-marin devra réagir à ses irrégularités. Les algorithmes de suivi peuvent utiliser les données de plusieurs capteurs comme un sonar, un altimètre, un ADCP, une caméra pour effectuer la mission.

Mission n°5

Suivre le véhicule de surface (ASV) du NURC qui bougera lentement tout en restant à 1.5m en dessous de ce dernier. Une balise acoustique à une fréquence de 12KHz sera placée au centre de l'ASV. Les équipes peuvent choisir d'effectuer la mission en utilisant la balise acoustique ou bien en analysant les images provenant d'une caméra qui regarde vers le haut. Utiliser les deux techniques donnera lieu à un bonus de 200 points.

Mission n°6

Faire surface dans une zone désignée par les organisateurs. Cette position sera envoyée au sous-marin via un modem acoustique (fréquence 20-24KHz). Un tel modem sera fourni par les organisateurs et les membres de l'équipe seront amenés à connecter le dispositif sur le sous-marin, l'alimenter et écrire le logiciel qui le gère. **La mission de surfacage doit être effectuée en fin du parcours.**

Obligations

Les équipes devront fournir un fichier log au maximum 10 minutes après le parcours. Le format du fichier log est un format ASCII avec la virgule comme séparateur avec comme données loguées : temps, position, action, commentaire entre guillemets (SSSS,XXX.x,YYY.y,ZZZ.z,AA.aa). Pour la mission du suivi de l'ASV, la distance et l'angle de vue du pinger devront être logués dans un fichier séparé.

Nos deux robots

Pour participer au concours nous disposons à ce jour de deux robots sous-marins *SAUC'ISSE* et *SARDINE* présentés sur la figure ci-dessous.

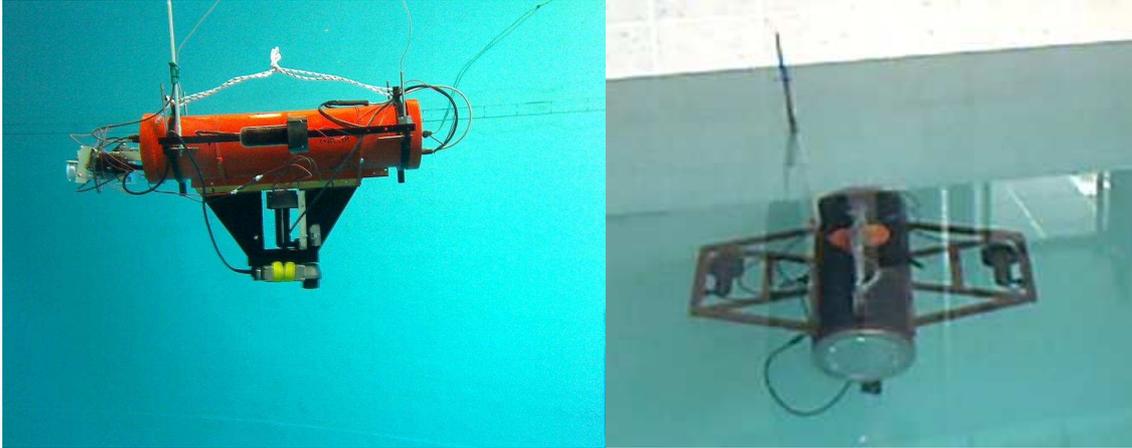


Figure 3 : SAUC'ISSE (à gauche) et SARDINE (à droite)

Pour effectuer les missions les robots sont munis de plusieurs capteurs (voir Figure 4):

- Caméra frontale : utilisée pour la mission de la bouée à libérer
- Caméra verticale : utilisée pour la mission pipeline
- Capteur de pression : utilisé pour connaître sa profondeur
- Boussole : utilisé pour la navigation
- Sonar et boussole : utilisé pour la localisation

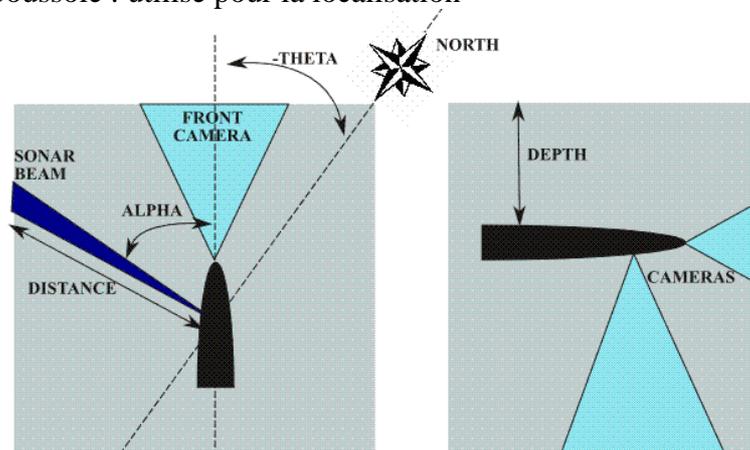


Figure 4 : Différents capteurs du robot

En analysant les missions du concours, on peut s'apercevoir que la localisation facilite considérablement l'exécution des missions ainsi que leur enchaînement. Par exemple si on commence le parcours dans la zone *Start1* (bonus de 100 points) alors il faudra rejoindre la zone *Start2* en utilisant la localisation. A la fin du parcours, il faut se localiser pour pouvoir rejoindre le point de surfacage dont les coordonnées sont fournies par les organisateurs... Dans la partie suivante, nous détailleront un peu plus l'algorithme de localisation que nous avons implémenté sur le sous-marin *SAUC'ISSE* (le seul ayant un sonar à ce jour).

Localisation

Expérience de localisation

Avant d'implémenter l'algorithme de localisation sur notre sous-marin, nous avons validé l'approche en utilisant des données enregistrées lors d'une expérience effectuée par un robot sous-marin d'une autre université. L'expérience à considérer ici a été conçue à la base pour illustrer le fonctionnement d'un algorithme de SLAM sous-marin [Ribas et al., 2008]. Les données ont été recueillies au cours d'une étude approfondie d'un port de plaisance abandonné dans la Costa Brava (Espagne). L'AUV (sous-marin autonome) de L'Université de Gironne a enregistré un jeu de données le long d'une trajectoire de 600 m qui comprenait une petite boucle autour du réservoir principal et une droite de 200 m à travers un canal étroit. L'ensemble de données comprenait des mesures d'un sonar sectoriel (Miniking Tritech), un DVL - Doppler Velocity Log - (Argonaut SonTek) et d'une centrale inertielle (Xsens MTI). Pour pouvoir valider le résultat du traitement de ces données, l'AUV a navigué près de la surface et a été muni d'un GPS qui a enregistré la trajectoire réelle du robot (vérité terrain). La Figure 5 montre un échantillon de données sonar, les points verts indiquent que le faisceau sonar a trouvé un obstacle (qui est en fait un mur de la marina). Comme on peut le voir, les données sont bruitées et contiennent beaucoup de valeurs aberrantes. Les longs segments signifient que les obstacles sont hors de portée du sonar (c.à.d. au-delà de 50m).

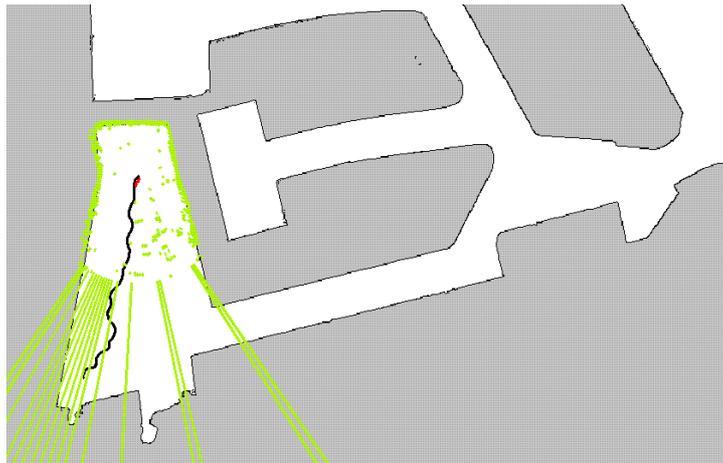


Figure 6 : Données du sonar prise dans une marina dans la Costa Brava

Mise en équation du système

Considérons un système caractérisé par les équations d'état suivantes :

$$\mathbf{f}_k : \mathbb{R}^{??} \rightarrow \mathbb{R}^{??}, \mathbf{g}_k : \mathbb{R}^{??} \rightarrow \mathbb{R}^l$$

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k)$$

$$\mathbf{y}_k = \mathbf{g}_k(\mathbf{x}_k).$$

Dans notre cas, \mathbf{x}_k est la position du robot à l'instant k . \mathbf{f}_k caractérise la dynamique du robot et \mathbf{y}_k est le vecteur de sortie qui est ce qu'on mesure (ici les mesures sonar). \mathbf{y}_k et \mathbf{x}_k sont reliés par la fonction d'observation \mathbf{g}_k qui exprime des relations géométriques entre la position, les mesures et la carte. On notera par \mathbb{X}_i et \mathbb{Y}_k les ensembles contenant \mathbf{x}_k et \mathbf{y}_k

respectivement. En utilisant la formule récursive des équations d'états, nous obtenons le système d'équations suivant :

$$\left\{ \begin{array}{l} \mathbf{x}_{k+1} = \mathbf{f}_k \circ \mathbf{g}_k^{-1}(\mathbf{y}_k) \\ \mathbf{x}_{k+1} = \mathbf{f}_k \circ \mathbf{f}_{k-1} \circ \mathbf{g}_{k-1}^{-1}(\mathbf{y}_{k-1}) \\ \dots \\ \mathbf{x}_{k+1} = \mathbf{f}_k \circ \mathbf{f}_{k-1} \circ \dots \circ \mathbf{f}_{k-n} \circ \mathbf{g}_{k-n}^{-1}(\mathbf{y}_{k-n}) \\ \mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) \\ \mathbf{x}_{k+1} \in \mathbb{R}^m, \mathbf{x}_k \in \mathbb{X}_k, \mathbf{y}_i \in \mathbb{Y}_i, i \in [k-n..k]. \end{array} \right.$$

Nous cherchons à connaître \mathbf{x}_{k+1} en fonction de l'état précédent \mathbf{x}_k à partir de plusieurs mesures $\mathbf{y}_i, i \in [k-n..k]$.

Comme nous allons le voir dans la partie suivante, un tel système d'équations se résout facilement avec les méthodes ensemblistes.

Utilisation de méthodes ensemblistes pour la résolution de système d'équations.

Résolution normale

Soit un système d'équations sous la forme :

$$\left\{ \begin{array}{l} \mathbf{f}_0(\mathbf{x}) = 0 \\ \mathbf{f}_1(\mathbf{x}) = 0 \\ \dots \\ \mathbf{f}_n(\mathbf{x}) = 0 \end{array} \quad \begin{array}{l} \mathbf{f}_i : \mathbb{R}^m \rightarrow \mathbb{R}^{n_i}, \\ \mathbf{x} \in \mathbb{X}. \end{array} \right.$$

Résoudre ce système d'équations revient à trouver l'ensemble des points qui satisfont toutes les équations i.e.

$$\mathbb{S} = \{\mathbf{x} \in \mathbb{R}^m, \forall i \in [0..n], \mathbf{f}_i(\mathbf{x}) = 0\}$$

Les méthodes ensemblistes permettent la manipulation d'ensembles (voir [Moore, 1979] et [Jaulin et al. 2001]). Par exemple, ces méthodes permettent de calculer une intersection d'ensembles $\mathbb{A} = \mathbb{B} \cap \mathbb{C}$, calculer une union $\mathbb{A} = \mathbb{B} \cup \mathbb{C}$, faire de l'inversion ensembliste $\mathbb{A} = f^{-1}(\mathbb{B})$, calculer l'image d'un ensemble par une fonction $\mathbb{A} = f(\mathbb{B})$...

Pour résoudre notre système d'équations, nous allons donc calculer des sous ensembles solutions pour chaque équation. Nous obtenons donc pour chaque équation

$$\mathbb{X}_i = \mathbf{f}_i^{-1}(0).$$

Puisqu'on cherche les points qui appartiennent à tous les sous-ensembles solutions, la solution finale recherchée sera donc sous la forme suivante

$$\mathbb{S} = \bigcap_{i \in \{0..n\}} \mathbb{X}_i$$

Résolution relaxée dans le cas de données aberrantes

Dans le cas de données aberrantes, certaines équations du système d'équation du robot qu'on souhaite résoudre ne seront pas satisfaites car elles se basent sur ces données mesurées.

Il faudra donc trouver un moyen de ne pas prendre en compte ces « fausses » équations. Le problème est qu'on ne sait pas quelles équations sont fausses.

La résolution normale qui fait l'intersection de tous les sous ensembles solutions X_i nous donnera un l'ensemble vide comme solution. Ceci est consistant car il n'y a aucun point qui satisfait toutes les équations puisque certaines d'entre elles sont fausses.

Les méthodes ensemblistes nous apportent la solution avec un autre type d'intersection appelé l'intersection q-relaxée exprimée par

$$S_q = \bigcap_{i \in \{0..n\}}^{\{q\}} X_i = \{x \in \mathbb{R}^m, \exists \mathbb{I} \subset \{1, \dots, n\}, \\ \text{card}(\mathbb{I}) = n - q, \forall i \in \mathbb{I}, x \in X_i\}$$

Un exemple avec 5 ensembles X_1, X_2, X_3, X_4, X_5 est représenté ci-dessous.

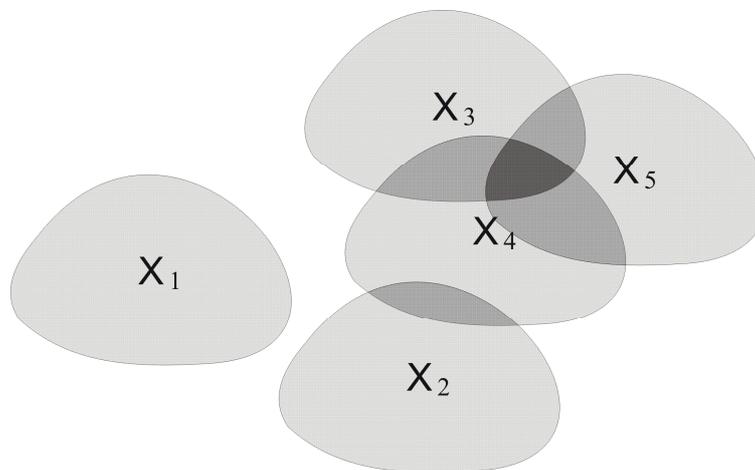


Figure 7 : Exemple de l'intersection relaxée de 5 ensembles

Dans l'exemple nous avons

$$\bigcap_{i \in \{0..5\}}^{\{0\}} X_i = \bigcap_{i \in \{1..5\}} X_i = \emptyset$$

$$\bigcap_{i \in \{0..5\}}^{\{2\}} X_i = X_3 \cap X_4 \cap X_5 \quad \text{La partie en gris foncé}$$

$$\bigcap_{i \in \{0..5\}}^{\{5\}} X_i = \bigcup_{i \in \{1..5\}} X_i$$

Résultats de la localisation dans la marina

La Figure 6 montre une comparaison entre la trajectoire GPS de référence (en noir) et la trajectoire (en bleu) obtenue par la méthode Dead Reckoning - estimation aveugle - qui est obtenue par fusion de données. Nous pouvons observer que la trajectoire obtenue par la méthode Dead Reckoning souffre d'une dérive non négligeable notamment en indiquant des

positions à l'extérieur du canal. La trajectoire calculée avec les méthodes ensemblistes est représentée sur la Figure 8. Notre algorithme retourne la trajectoire sous forme de boîtes (en rose), mais nous prenons habituellement le centre de ces boîtes comme position réelle (en rouge). La trajectoire rouge suit la trajectoire GPS. L'exécution de l'algorithme est temps réel sur un ordinateur portable avec un processeur Intel Core 2 Duo.

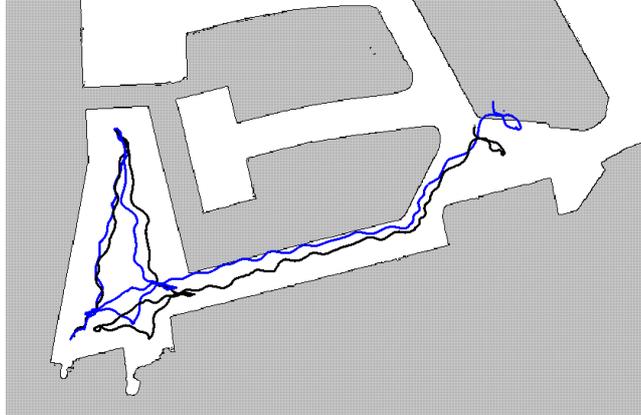


Figure 9 : Comparaison entre le Dead Reckoning et la trajectoire GPS de référence

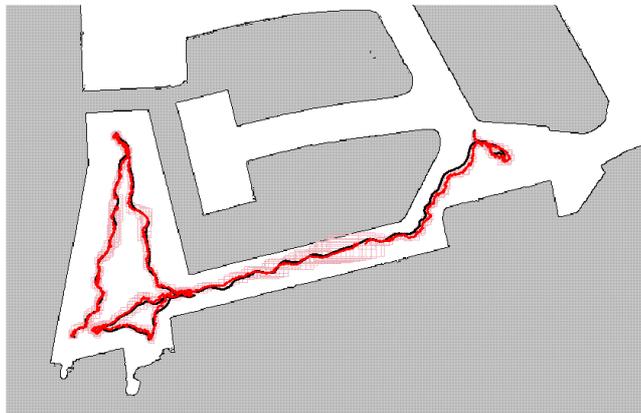


Figure 10 : Comparaison entre notre méthode et la trajectoire GPS de référence

Références

- [Ribas et al., 2008] D. Ribas, P. Ridao, J.D. Tardós, and J. Neira. Underwater SLAM in man made structured environments. *Journal of Field Robotics*, Accepted for publication, 2008.
- [Moore, 1979] R. E. Moore. *Methods and Applications of Interval Analysis*. SIAM, Philadelphia, PA, 1979.
- [Jaulin et al., 2001] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied Interval Analysis, with Examples in Parameter and State Estimation, Robust Control and Robotics*. Springer-Verlag, London, 2001.