

Bureau d'étude : SLAM

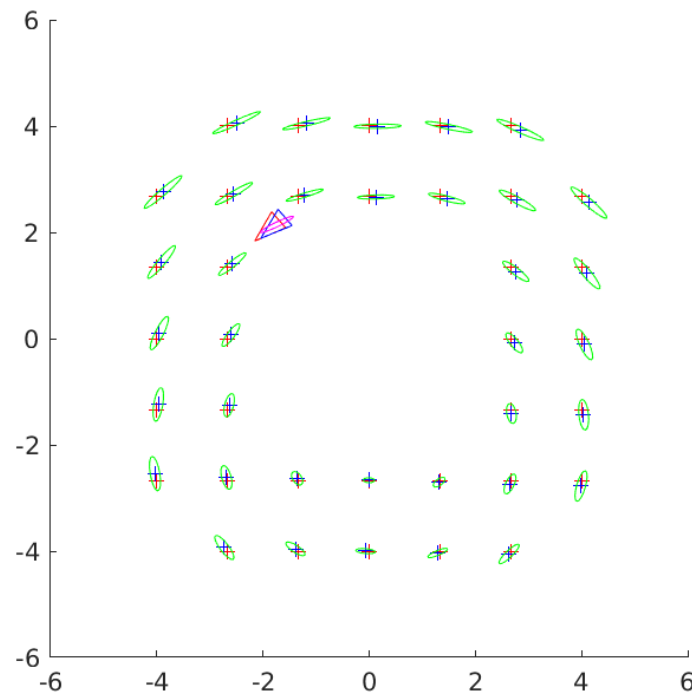
Simultaneous Localization And Mapping

Dans un premier temps, on paramètre l'algorithme SLAM afin d'obtenir des résultats proches de ceux obtenus avec le script *slam2d*. Ces réglages serviront de références pour l'analyse des performances de l'algorithme.

On choisit donc le réglage suivant pour le filtre de Kalman :

- $q = [q_x = 0.01 \quad q_\alpha = 1 * \frac{\pi}{180}]$;
- $s = [s_d = 0.1 \quad s_\alpha = 3 * \frac{\pi}{180}]$;
- Covariance nulle sur la position initiale du robot.

On obtient le résultat suivant à la fin de la simulation :

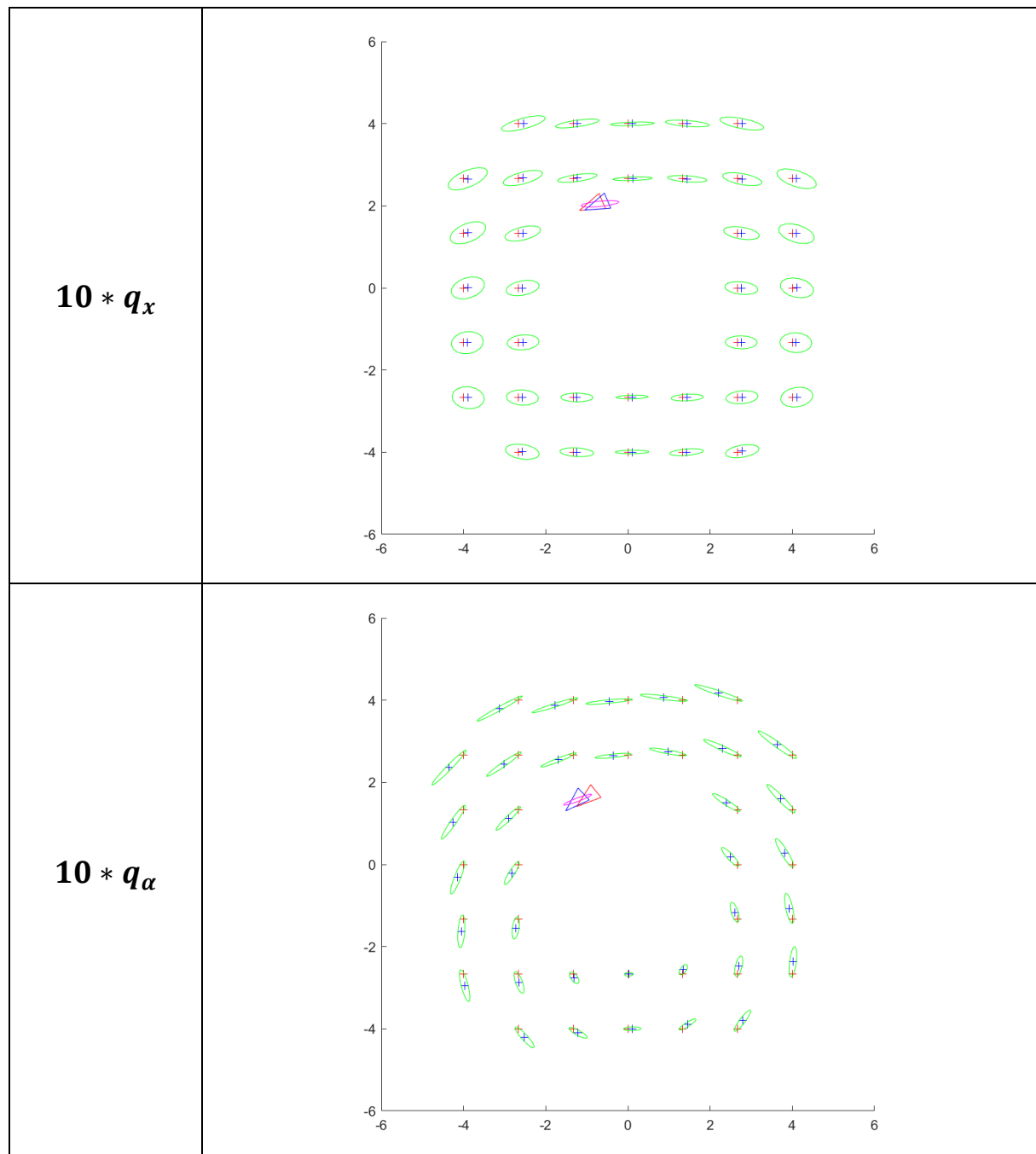


On constate que la position estimée du robot est très fidèle même après que celui-ci a effectué plusieurs tours. De plus la position estimée des amers est également associée à une incertitude assez faible ce qui permet de reconstruire une carte fidèle de l'environnement d'évolution du robot.

Analyse de l'influence du bruit d'évolution

Le premier paramètre que nous étudions est le bruit d'évolution. Celui-ci traduit l'incertitude introduite au moment de l'estimation de l'état du système.

Afin d'analyser l'impact de ce paramètre sur les performances globales de la méthode SLAM, nous réalisons plusieurs simulations en augmentant successivement le bruit d'évolution lié à la translation du robot (q_x) et celui lié à la rotation (q_α).

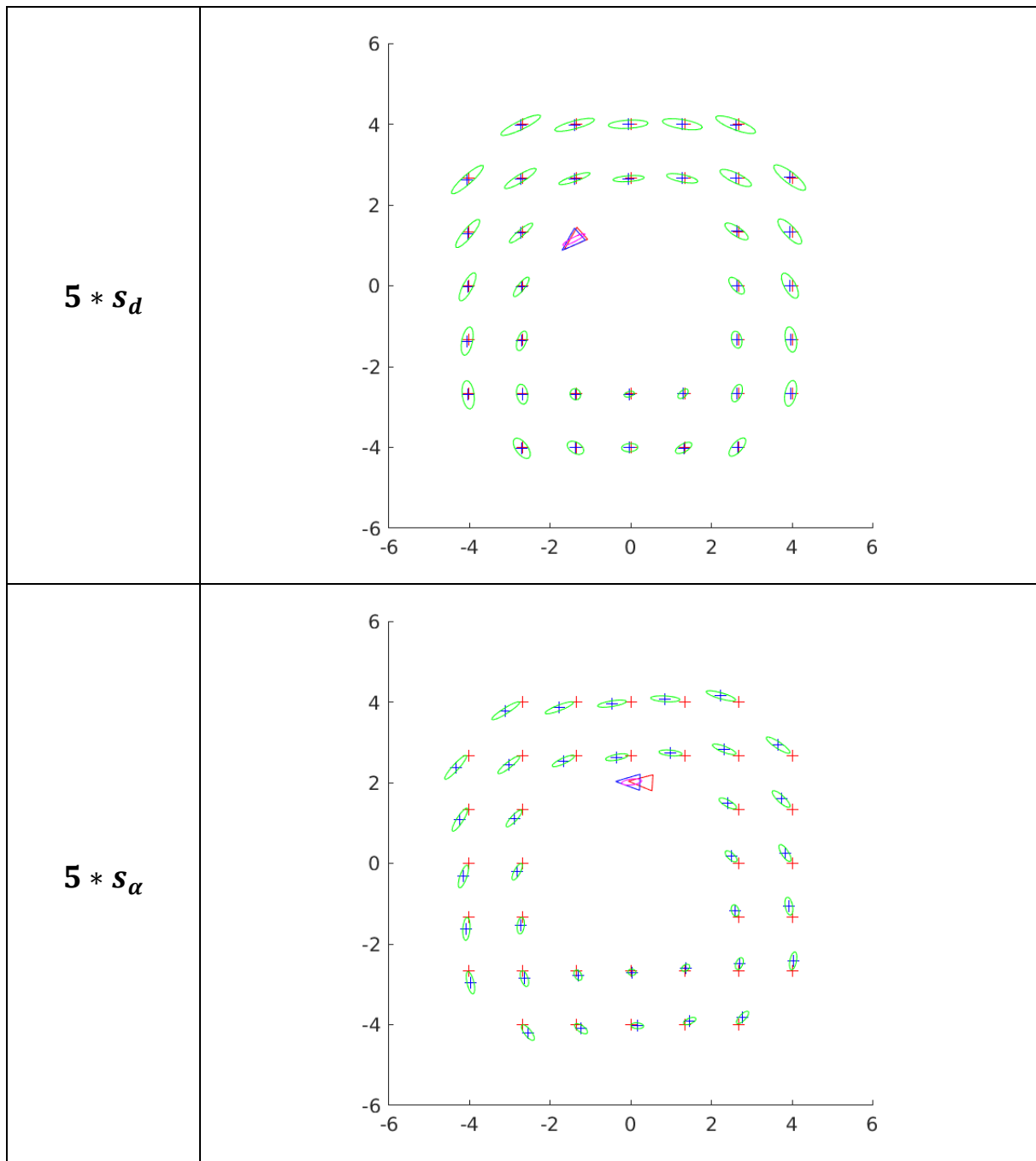


On constate que l'estimation de la position du robot est moins précise. De plus, on constate que l'incertitude sur la position des amers augmente (les ellipses s'allongent selon les tangentes au cercle centré sur la trajectoire pour le bruit angulaire et selon les rayons pour le bruit en distance). En effet, même si dans ce cas le bruit de mesure est faible (on considère les mesures comme très fiables), la position absolue des amers ne peut pas être déterminée précisément à cause de l'incertitude existant sur la position du robot.

Il semble donc que contrairement à un filtre de Kalman classique, le SLAM ne peut pas compenser un fort bruit d'évolution par des mesures très précise. Ceci est lié au fait que les mesures effectuées par le robot ne donnent pas directement une indication de la position absolue de celui-ci dans son environnement.

Analyse de l'influence du bruit de mesure

On analyse ensuite l'influence du bruit de mesure (incertitude sur la précision des mesures effectuées par le robot) en procédant comme pour la partie précédente.



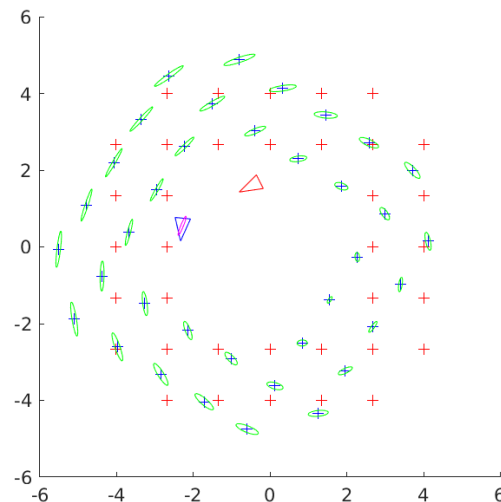
Les incertitudes sur la position des amers sont importantes au départ mais diminuent rapidement. Au final, les biais sur les positions estimés sont assez faibles (en effet contrairement au cas précédent, seuls les bruits de mesures viennent réellement perturber les estimations).

Analyse de l'influence des conditions initiales

On analyse également l'influence des paramètres liés au positionnement initial du robot.

Position initiale

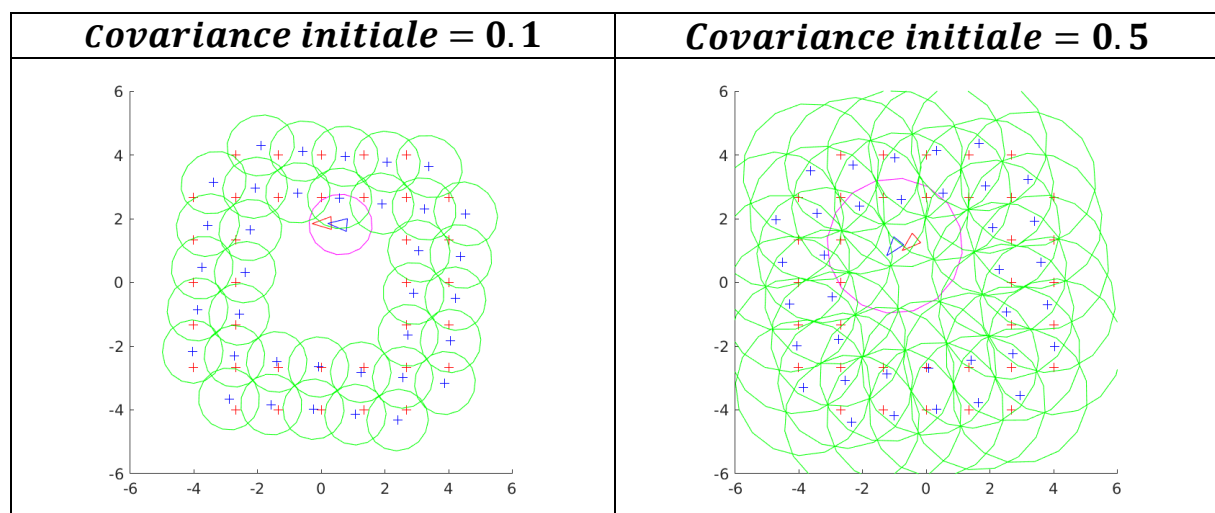
On effectue une simulation en changeant la valeur initiale de la position du robot mais en gardant une covariance nulle.



Dans ce cas, la carte sera biaisée mais correcte (l'écart entre l'estimation et la position réelle du robot correspondra à chaque instant à l'écart introduit initialement). Comme le robot ne dispose d'aucun moyen direct de mesure sa position, l'écart initial ne peut pas être rattrapé grâce aux mesures, aussi précises soient-elles.

Incertitude sur la position initiale

On effectue également une simulation en augmentant la covariance sur la position initiale du robot.



Les incertitudes sur la position du robot et des amers augmentent dans ce cas. Encore une fois, parce que le robot ne dispose d'aucune mesure absolue de position, il n'est pas possible de diminuer l'incertitude initiale sur sa position. La faible précision sur la localisation initiale du robot introduit un biais dans la cartographie de l'environnement.

Calcul de l'angle de l'innovation

Après avoir calculer l'innovation, on effectue le calcul suivant :

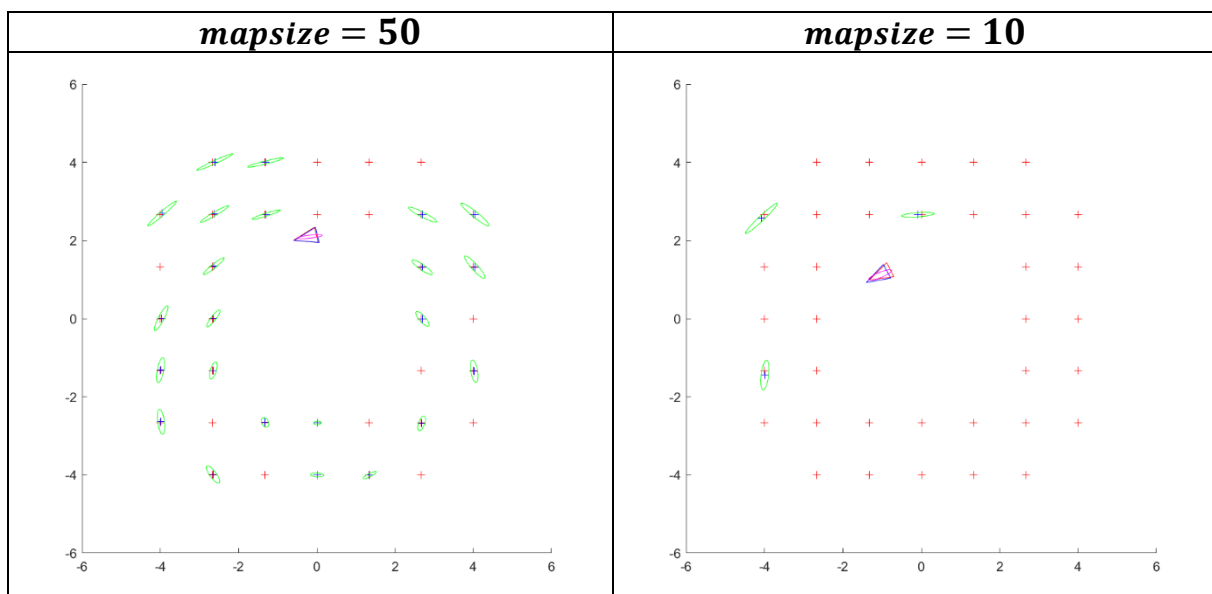
```
if z(2) > pi
    z(2) = z(2) - 2*pi;
end
if z(2) < -pi
    z(2) = z(2) + 2*pi;
end
```

Ce code permet de s'assurer que l'angle de l'innovation est centré en 0 et contenu dans l'intervalle $[-\pi; \pi]$. Cette vérification est nécessaire car un angle supérieur à 2π représente un écart numérique qui ne correspond pas à la réalité géométrique de la situation. La correction du filtre de Kalman ne serait alors pas adaptée car un écart angulaire en réalité très petit pourrait être considéré comme numériquement grand.

Analyse de l'influence de la taille de la carte

Le dernier paramètre dont nous analysons l'influence est la taille mémoire dont nous disposons pour stocker les positions des différents amers.

On réalise donc une série de simulation en diminuant à chaque fois la taille de la carte.



La première chose à remarquer est que, si la carte est trop petite, le robot ne peut pas enregistrer la position de tous les amers qu'il rencontre. La carte sera donc nécessairement moins détaillée si on réduit la quantité d'information que peut stocker le robot.

Les performances sont peu affectées par la réduction de la taille de la carte. Si on prend des bruits d'états et de mesures faibles, le robot parvient à se situer même en enregistrant la position de seulement quelques amers. En effet, comme le robot dispose d'un moyen d'identification des amers, les amers non enregistrés ne gênent pas la localisation. De plus, comme il effectue un circuit de déplacement assez petit, il repasse régulièrement devant les mêmes points.

Axes d'amélioration

Dans l'optique d'adapter ce code à une situation réelle, plusieurs éléments du code pourrait être modifiés.

Adaptation du bruit de mesure

Pour commencer il faudrait utiliser notre connaissance physique des capteurs afin de régler le bruit de mesure. En effet le bruit de mesure utilisé pour le filtre de Kalman doit être un bruit blanc Gaussien qui englobe les bruits réels des capteurs. Prendre un bruit trop faible entraînerait un écart entre l'estimation et la position réelle car on accorderait trop d'importance à des mesures bruitées. Prendre un bruit trop fort ne permettrait pas d'utiliser de manière optimale les informations transmises par les capteurs.

Reconnaissance des amers

Une autre partie du programme SLAM qu'il faudrait adapter dans le cadre d'une utilisation réelle est la reconnaissance des amers.

Dans la simulation, la fonction permettant de mesurer la distance à un amer prend en argument l'ID de cet amer. Ceci suppose que nous disposons d'un moyen d'identifier les amers observés de manière sûre, ce qui n'est pas vrai en général. Le plus souvent, c'est le robot qui doit réaliser cette étape de reconnaissance.

Il faudrait donc ajouter dans le code une étape de reconnaissance des amers durant laquelle le robot utilise des informations visuelles, de positionnement et des informations topologique pour déterminer si l'amer a déjà été observé. La boucle ne se ferait alors pas sur tout les amers connus mais sur les amers qui peuvent être observé à chaque instant avec pour chaque amer une phase d'identification visant à déterminer si celui-ci est connu où s'il s'agit d'un nouvel élément de la carte.

Ajout de mesures supplémentaires

Le dernier axe d'amélioration pour une utilisation réelle serait d'ajouter des capteurs permettant de mesurer la position du robot. Par exemple l'utilisation d'un GPS permettrait d'obtenir des indications sur la position absolue du robot ce qui permettrait de corriger d'éventuels biais.

Pour utiliser le GPS, il faudrait ajouter, à chaque fois quelles sont transmises, les informations de localisation dans le calcul de l'innovation. Ceci permettrait de prendre en compte la mesure absolue de la position afin de périodiquement réduire les incertitudes de positionnement du robot.