Robótica

Exercise 3.3. Differential motion of a robot with **odometry** commands

En el ejercicio 3.2 se implementó el movimiento del robot mediante comandos de velocidad (linear y angular). Ahora vamos a hacer lo propio empleando el modelo de movimiento basado en odometría, esto es $u_t = (\Delta x \; \Delta y \; \Delta\theta)'$. Lo implementaremos tanto en su forma analítica, como muestreada.

1.- Forma analítica. Mover un robot a lo largo del cuadrado del ejercicio 3.1 (8x8 metros), con incrementos de pose de 2 metros, y dibujar la elipse de incertidumbre sobre la pose odométrica (donde cree que está) y también una marca sobre la posición real, generada aleatoriamente a partir de la matriz covarianzas de un ruido gaussiano.

$$\Sigma_{\Delta p} = \begin{bmatrix} 0.04 & 0 & 0 \\ 0 & 0.04 & 0 \\ 0 & 0 & 0.01 \end{bmatrix}$$

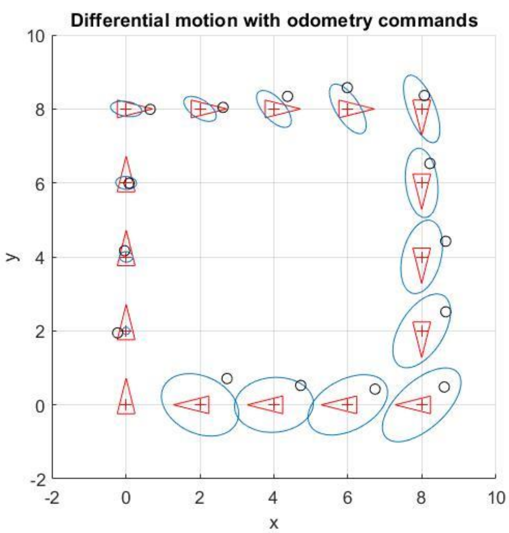

Figure 1

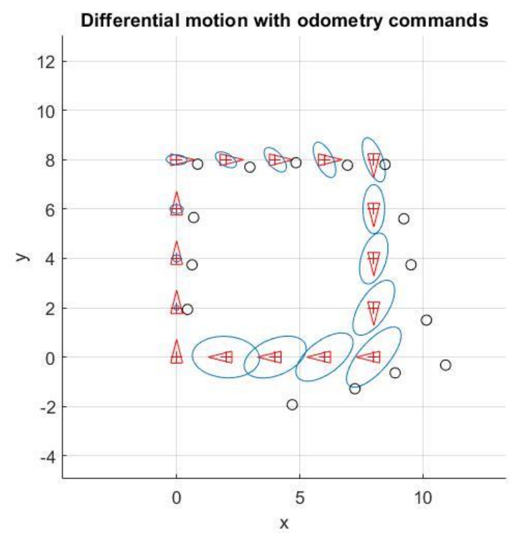

Figure 2

As we see in both figures, this is the representation of the ideal path of a robot using odometry, the real position of the robot (represented by a circular point) and an ellipse which represent the possible location where can be the robot. This ellipse follows a gaussian distribution which change in every step of the robot and the sigma depends where the robot was in his real location.

2.- Forma muestreada. Si generamos muestras aleatorias de la distribución normal del apartado anterior obtendríamos poses que no son factibles con el movimiento no-holonómico del robot, esto es, no se corresponden con comando v, w con ruido. Una manera más realista es generar las muestras a partir del modelo $u_t = (\theta_1 \; d \; \theta_2)'$, es decir, modelando un giro, avance y giro. Estos parámetros se obtienen fácilmente de los valores odométricos $[\hat{x}_t, \hat{y}_t, \hat{\theta}_t]$ y $[\hat{x}_{t-1}, \hat{y}_{t-1}, \hat{\theta}_{t-1}]$:

$$\begin{aligned}
\theta_1 &= \text{atan2}(\hat{y}_t - \hat{y}_{t-1}, \hat{x}_t - \hat{x}_{t-1}) - \hat{\theta}_{t-1} \\
d &= \sqrt{(\hat{x}_t - \hat{x}_{t-1})^2 + (\hat{y}_t - \hat{y}_{t-1})^2} \\
\theta_2 &= \hat{\theta}_t - \hat{\theta}_{t-1} - \theta_1
\end{aligned}$$

- Implementa una función, que dados $[\hat{x}_t, \hat{y}_t, \hat{\theta}_t]$ y $[\hat{x}_{t-1}, \hat{y}_{t-1}, \hat{\theta}_{t-1}]$ calcule $u_t = (\theta_1 \; d \; \theta_2)'$.
- Utilizando esta acción de control $u_t = (\hat{\theta}_1 \; \hat{d} \; \hat{\theta}_2)'$, se puede modelar la acción ruidosa como:

$$\begin{aligned}
\theta_1 &= \hat{\theta}_1 + \text{sample}(\alpha_1 \, \hat{\theta}_1^2 + \alpha_2 \, \hat{d}^2) \\
d &= \hat{d} + \text{sample}(\alpha_3 \, \hat{d}^2 + \alpha_4 \, (\hat{\theta}_1^2 + \hat{d}^2)) \\
\theta_2 &= \hat{\theta}_2 + \text{sample}(\alpha_1 \, \hat{\theta}_2^2 + \alpha_2 \, \hat{d}^2)
\end{aligned}$$

  Donde *sample*(b) es un valor aleatorio que sigue una distribución N (0, b) y α_i modelan el ruido intrínseco del robot. Utilizando este modelo basado en giro-avance-giro, la posición del robot se puede calcular como:

$$\begin{aligned}
x_t &= x_{t-1} + d\cos(\theta_{t-1} + \theta_1) \\
y_t &= y_{t-1} + d\sin(\theta_{t-1} + \theta_1) \\
\theta_t &= \theta_{t-1} + \theta_1 + \theta_2
\end{aligned}$$

Dibuje la posición del robot (el ángulo o *heading* no se dibuja) en base a partículas (*samples* de poses probables en las que se puede encontrar el vehículo). Considera #*particles*=100 y los siguientes valores de α_i: `a1=0.07; a2=0.07; a3=0.03; a4=0.05.` Juega con diferentes valores de α_i. Para mejorar la visualización de los resultados, en esta ocasión considera que el robot se mueve con incrementos de 0,5m, pero sólo muestra la posición de las partículas cada 4 incrementos.

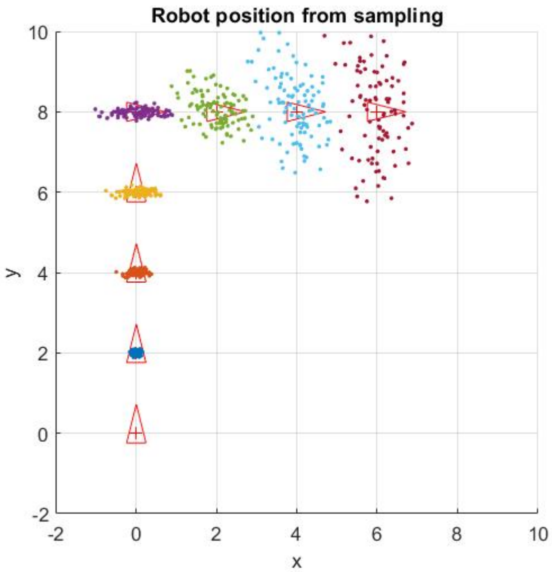

Figure 3. Robot position with a1=0.07, a2=0.07, a3=0.03 and a4=0.05

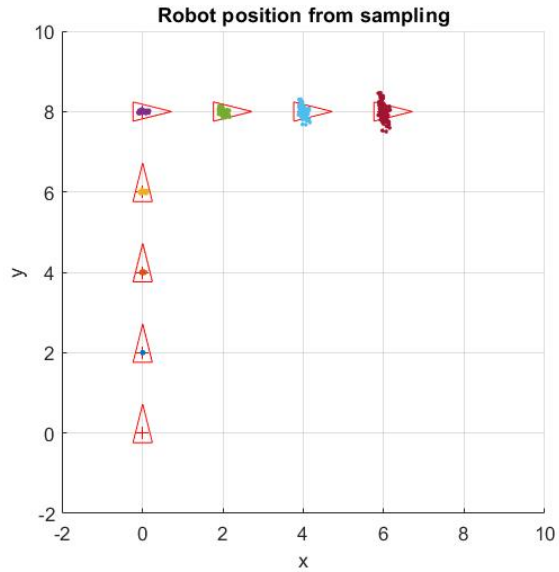

Figure 4. Robot position with a1=0.01, a2=0.01, a3=0.01 and a4=0.01

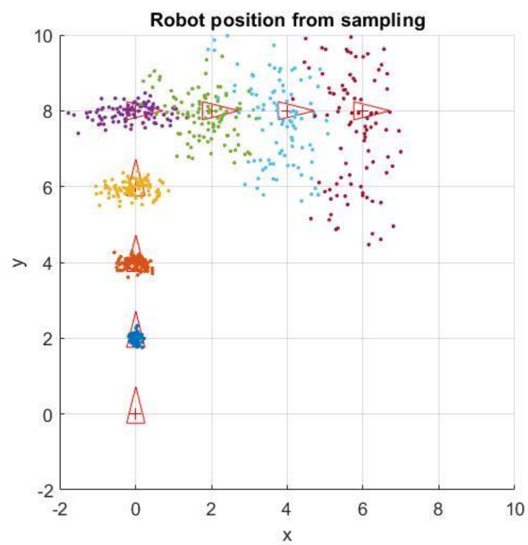

Figure 5. Robot position with a1=0.1, a2=0.1, a3=0.1 and a4=0.1

These figures represent the ideal path of the robot and the real positions with can be the robot (the filled points). For the points, we made a N samples with will have alphas for the gaussian error and follow a gaussian distribution which, for each step, they are more dispersed, and this means that the gaussian distribution increases in his sigma. If these alphas have values so low, the points are almost close of each other, which means it has no much error. However, if these have high values, the points are more dispersed, and thus it has a noticeable error.