Autonomous Navigation and Perception (086762)
Homework #2

Submission guidelines:

- Submission in pairs by: 19 April 2021, 14:30.
- Email submission as a *single zip or pdf file* to `anpsubmission@gmail.com`.
- Make sure to name the submitted file ID1-ID2.zip (or .pdf) with ID1,ID2 - the students' ids.
- For hands-on parts, your source code should be submitted as well (include it in the above zip file).

Theoretical questions:

1. Consider a planning session at time instant 0 and a given set of non-myopic candidate actions $\mathcal{A} \doteq \{a_{0:L-1}\}$, where L is the planning horizon. Denote the belief over state X_k at time instant k by $b_k \doteq \mathbb{P}(X_k \mid a_{0:k-1}, z_{1:k})$, and assume motion and observation models, $\mathbb{P}(X_k \mid X_{k-1}, a_{k-1})$ and $\mathbb{P}(z \mid X)$, are given. Consider an objective function $J(.,)$ of expected cumulative rewards given some reward (cost) function $r(b, a)$.
   (a) Write the objective function explicitly for the above-considered setting.
   (b) Derive a recursive formulation that expresses $J(b_0, a_{0:L-1})$, for a given non-myopic action $a_{0:L-1} \in \mathcal{A}$, via $J(b_1, a_{1:L-1})$. Present the derivation in detail.
   (c) Consider now the corresponding value function $V_0^\pi(b_0)$ and some stationary policy π within the same setting.
      i. Write an explicit expression for the value function $V_0^\pi(b_0)$.
      ii. Similar to clause 1b, derive a recursive formulation that expresses $V_0^\pi(b_0)$ in terms of $V_1^\pi(b_1)$.
      iii. Discuss the difference between $J(b_0, a_{0:L-1})$ and $V_0^\pi(b_0)$.

Hands-on tasks: You are highly encouraged to use the Julia language for all hands-on tasks. However, this is not compulsory. In case you use Julia, the supplied skeleton may be helpful.

1. Consider a mobile robot navigating in a 2D environment. For simplicity, we consider only the position state of the robot, i.e. $X_i \in \mathbb{R}^2$ denotes the x-y robot position at time instant i. The motion and observation models, $\mathbb{P}(X_{k+1} \mid X_k, a_k)$ and $\mathbb{P}(z_k \mid X_k)$, are given by

$$X_{k+1} = f(X_k, a_k) + w \quad , \quad z_k = h(X_k) + v \tag{1}$$

   where w and v are process and measurement noise terms, with $w \sim N(0, \Sigma_w)$ and $v \sim N(0, \Sigma_v)$.

   In this exercise, we shall consider simple linear models, i.e. $f(X_k, a_k) = F \cdot X_k + a_k$ with $F = I_{2\times 2}$ and $a_k \in \mathbb{R}^2$ is the commanded position displacement. We also consider position measurements such that $z_k \in \mathbb{R}^2$ and $h(X_k) \equiv X_k$.

   Consider a prior over robot position at time instant 0 is available: $b(X_0) \doteq \mathbb{P}(X_0) = N(\mu_0, \Sigma_0)$.

   (a) Implement a function `PropagateUpdateBelief`, which updates a Gaussian belief according to a (given) performed action and captured observation, considering a recursive setting.
      - Input: a Gaussian belief from some time instant k over X_k, i.e. $b(X_k) = N(\mu_k, \Sigma_k)$, action a_k and observation z_{k+1}.
      - Output: an updated posterior Gaussian belief at the next time step, i.e. $b(X_{k+1}) = N(\mu_{k+1}, \Sigma_{k+1})$.

(b) Implement a function `SampleMotionModel` that, given a robot state X and action a, generates the next state, i.e. $X' \sim \mathbb{P}(X' \mid X, a)$, according to the motion model in Eq. (1).

- Input: State X, action a
- Output: Next state X'

(c) Implement a function `GenerateObservation`, which generates an observation according to the observation model in Eq. (1), i.e. $z \sim \mathbb{P}(z \mid X)$.

- Input: State X
- Output: Observation z

(d) Consider $b(X_0) = N(\mu_0, \Sigma_0)$ with $\mu_0 = (0,0)^T$ and $\Sigma_0 = I_{2\times 2}$, $\Sigma_w = 0.1^2 \cdot I_{2\times 2}$, $\Sigma_v = 0.01^2 \cdot I_{2\times 2}$, and $a_k = (1,0)^T$. Let the actual initial robot location (ground truth) be $X_0 = (0.5, 0.2)^T$.

i. Generate a possible robot trajectory $\tau \doteq \{X_0, X_1, \ldots, X_T\}$ where the robot starts at X_0 and follows a given action sequence $a_{0:T-1} \doteq \{a_0, a_1, \ldots, a_{T-1}\}$. Assume $T = 10$, and $a_i = (1,0)^T$ for $i \in [0, T-1]$.

ii. Generate observations along the trajectory τ, by generating a single z_i for each $X_i \in \tau$.

iii. Calculate belief propagation along the trajectory τ while only considering a motion model (without observations). In other words, for each time step $i \in [0, T]$, calculate $\mathbb{P}(X_i \mid a_{0:i-1}, b(X_0)) \doteq N(\mu_i^{mm}, \Sigma_i^{mm})$, considering the prior belief $b(x_0)$ and the probabilistic motion model (mm). Draw on a plot the actual trajectory and these propagated beliefs[1] (in terms of mean and covariance) versus time.

iv. Calculate the *posterior* beliefs along the trajectory τ, this time also using observations (generated observations in 1(d)ii). In other words, for each time step $i \in [0, T]$, calculate $\mathbb{P}(X_i \mid a_{0:i-1}, z_{1:i}, b(X_0)) \doteq N(\mu_i, \Sigma_i)$, considering the prior belief $b(X_0)$ and the probabilistic motion and observation models. Draw on a separate plot the actual trajectory, and these posterior beliefs (in terms of mean and covariance) versus time.

2. We now modify the problem setting as follows. Several beacons are scatted in the environment. The locations of these beacons are known to the robot (denote it by X_j^b for the jth beacon), and represented by a matrix $X^b \in \mathbb{R}^{2\times n}$, i.e. $X^b = [X_1^b, \ldots, X_n^b]$, where n is the number of beacons.

   Assume the robot gets a relative position observation $z^{rel} \in \mathbb{R}^2$ from a beacon if it is sufficiently close to it (distance less than d). For simplicity, consider the beacons are positioned sufficiently apart such that an observation from only one of the beacons can be obtained for every possible robot location. Moreover, assume the accuracy of these observations deteriorates with the distance r to the beacon, starting from a certain given distance $r_{min} \leq d$, such that for $r \leq d$, the measurement covariance is $\Sigma_v = (0.01 \cdot \max(r, r_{min}))^2 \cdot I_{2\times 2}$.

(a) Write the corresponding observation model for the measurement z^{rel}, as a function of robot location X, beacon location X^b, and additional parameters mentioned above.

(b) Implement a function `GenerateObservationFromBeacons`, which generates an observation z^{rel} according to the observation model from clause 2a. Assume X^b, r, r_{min} and d are given.

- Input: Robot location X
- Output: Measurement z^{rel} and index of beacon within range; null otherwise.

(c) Consider $n = 9$ beacons equally scattered in a 9×9 grid, and let the coordinate of its bottom left corner be $(0,0)$. To clarify, one of the beacons is located at $(0,0)$. Assume $b(X_0) = N(\mu_0, \Sigma_0)$ with $\mu_0 = (0,0)^T$ and $\Sigma_0 = I_{2\times 2}$, $\Sigma_w = 0.1^2 \cdot I_{2\times 2}$. Let the actual initial robot location (ground truth) be $X_0 = (-0.5, -0.2)^T$. Consider an action sequence $a_{0:T-1} = \{a_0, \ldots, a_{T-1}\}$, with $a_i \doteq (0.1, 0.1)^T$, $i \in [0, T-1]$, for $T = 100$. Further, let $d = 1$ and $r_{min} = 0.1$.

i. Repeat clauses 1(d)i-1(d)iv. Indicate in all plots also the n beacons.

ii. Repeat clauses 1(d)i-1(d)iv, while considering a fixed measurement covariance $\Sigma_v = 0.01^2 I_{2\times 2}$ (measurement is still obtained only if $r \leq d$). Indicate in all plots also the n beacons.

iii. Present in a new figure localization estimation errors[2] $\tilde{X}$ versus time for the two cases above.

iv. Present in a new figure the square root of the trace of estimation covariance for the two cases above. Compare and analyze qualitatively the obtained results.

[1] You can use the supplied `drawCovarianceEllipse` function to draw a 1-std covariance ellipse. Alternatively, similar Julia function covellipse! from package StatsPlots.jl is available.

[2] $\tilde{X} = \hat{X} - X$, where $\hat{X}$ is an estimate and X is the ground truth value.

(d) Leveraging the above capabilities, let us now consider a simple *belief space planning* problem. Given a set of action sequences $\mathcal{A} = \{a^j_{0:T-1}\}$, the robot has to choose the one that is expected to yield minimum uncertainty at the final state (X_T). In this question we consider $\det(\Sigma)$ as the information-theoretic cost function $c(b, a)$.

Further, consider $N = 10$ simple action sequences in $\mathcal{A} = \{a^j_{0:T-1}\}_{j=1}^N$, where the jth action sequence $a^j_{0:T-1}$ is given by $a^j_{0:T-1} = \{a^j_0, \ldots, a^j_{T-1}\}$, where $a^j_i \doteq (0.1, 0.1 \cdot j/5)^T$.

i. Generate a single trajectory realization τ^j for each action sequence in $\mathcal{A} = \{a^j_{0:T-1}\}_{j=1}^N$ by following clause 1(d)i. Draw all N trajectories and indicate beacon locations on a single plot.

ii. Calculate the posterior beliefs along each trajectory τ^j, $\forall j \in [1, N]$, and evaluate the information theoretic cost. Plot the information-theoretic cost versus trajectory index. Designate the best action for the given cost, and explain why this result makes sense.