AMR HW6
R99922070 俞冠廷
2011/4/26

Ch7, Exercise 3.

You are requested to derive an EKF localization algorithm for a simplistic underwater robot. The robot lives in a 3-D space and is equipped with a perfect compass (it always knows its orientation). For simplicity, we assume the robot move independently in all three Cartesian directions (x, y, and z), by setting velocities $\dot{x}, \dot{y}, \dot{z}$. Its motion noise is Gaussian and independent for all directions.

The robot is surrounded by a number of beacons that emit acoustic signals. The emission time of each signal is known, and the robot can determine from each signal the identity of the emitting beacon (hence there is no correspondence problem). The robot also knows the location of all beacons, and it is given an accurate clock to measure the arrival time of each signal. However, the robot cannot sense the direction from which it received a signal.

(a) You are asked to devise an EKF localization algorithm for this robot. This involves a mathematical derivation of the motion and the measurement model, along with the Taylor approximation. It also involves the statement of the final EKF algorithm, assuming known correspondence.

(a) Underwater Robot EKF localization algorithm

Because the robot knows the exact emission time and arrival time of each signal, somehow we can infer the distance between the landmark and the robot. Therefore, in the measurement is a single value - range.

$$g : \bar{\mu}_t = \mu_{t-1} + \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \Delta t$$

$$G_t = \nabla g = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_t = \begin{bmatrix} {\sigma_x}^2 & 0 & 0 \\ 0 & {\sigma_y}^2 & 0 \\ 0 & 0 & {\sigma_z}^2 \end{bmatrix}$$

$$\bar{\Sigma}_t = G_t \Sigma_{t-1} {G_t}^T + R_t$$

for all observed features $z_t^i = r_t^i$ do

$$j = c_t^i$$

$$q = (m_{j,x} - \bar{\mu}_{t,x})^2 + (m_{j,y} - \bar{\mu}_{t,y})^2 + (m_{j,z} - \bar{\mu}_{t,z})^2$$

$$\hat{z}_t^i = \left[\sqrt{q}\right]$$

$$H_t^i = \begin{bmatrix} \frac{\bar{\mu}_{t,x} - m_{j,x}}{\sqrt{q}} & \frac{\bar{\mu}_{t,y} - m_{j,y}}{\sqrt{q}} & \frac{\bar{\mu}_{t,z} - m_{j,z}}{\sqrt{q}} \end{bmatrix}$$

$$S_t^i = H_t^i \bar{\Sigma}_t \left[H_t^i\right]^T + Q_t$$

$$K_t^i = \bar{\Sigma}_t \left[H_t^i\right]^T \left[S_t^i\right]^{-1}$$

$$\bar{\mu}_t = \bar{\mu}_t + K_t^i (z_t^i - \hat{z}_t^i)$$

$$\bar{\Sigma}_t = (I - K_t^i H_t^i) \bar{\Sigma}_t$$

endfor

$\mu_t = \bar{\mu}_t$

$\Sigma_t = \bar{\Sigma}_t$

(b) Implement your EKF algorithm and a simulation of the environment. Investigate the accuracy and the failure modes of the EKF localizer in the context of the three localization problems: global localization, position tracking, and the kidnapped robot problem.

```
function hw6
mu = [0 0 0]';
Sig = eye(3);
dt = 0.1;
sigx = 0.1;
sigy = 0.1;
sigz = 0.1;
nlmk = 20;
m = zeros(nlmk,3);
timestep = 100;
hitory = zeros(timestep,3);
```

```
gt = zeros(timestep,3);
for i =1:nlmk
    m(i,1) = rand()*60-10;
    m(i,2) = rand()*60-10;
    m(i,3) = rand()*60-10;
end



for k=2:timestep
v = [1 2 3]';
R = [(sigx*v(1))^2 0 0;
    0, (sigy*v(2))^2, 0;
    0, 0, (sigz*v(3))^2];

mub = mu + v * dt+[randn()*0.1, randn()*0.1, randn()*0.1]'; % add noise
G = eye(3);
Sigb = G*Sig*G' + R;
Q = 10^2;
gt(k,:) = (gt(k-1,:)+v'*dt);
plotUncertainEllip3D(Sigb, mub);
for i=1:nlmk
    j = i;
    q = (m(j,1) - gt(k,1))^2 + (m(j,2) - gt(k,2))^2 + (m(j,3) - gt(k,3))^2;
    qh = (m(j,1) - mub(1))^2 + (m(j,2) - mub(2))^2 + (m(j,3) - mub(3))^2;
    z = sqrt(q); % + randn()*0.1; % add noise
    zh = sqrt(qh);
    H = [mub(1)-m(j,1), mub(2)-m(j,2), mub(3)-m(j,3)]/sqrt(q);
    S = H*Sigb*H' + Q;
    K = Sigb*H'*inv(S);
    mub = mub + K * (z - zh);
    Sigb = (eye(3)-(K*H)) * Sigb;
end
hold on
plot3(m(:,1),m(:,2),m(:,3),'r.');
plotUncertainEllip3D(Sigb, mub);
hitory(k,:) = mub';
plot3(hitory(1:k,1),hitory(1:k,2),hitory(1:k,3),'b-');
plot3(gt(1:k,1),gt(1:k,2),gt(1:k,3),'k-');
```

```
axis equal
legend('uncertainty ellipsoid','estimated path','landmark');
hold off
pause(0.5);

mu = mub;
Sig = Sigb;
end
end
```

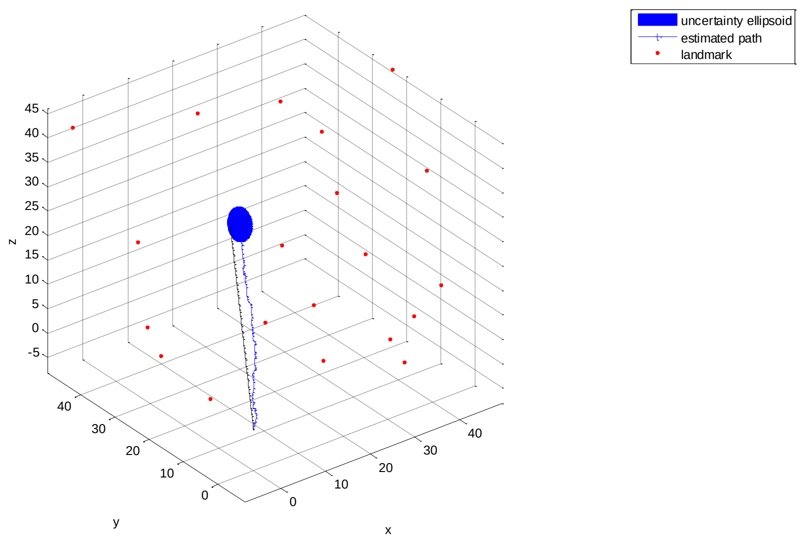
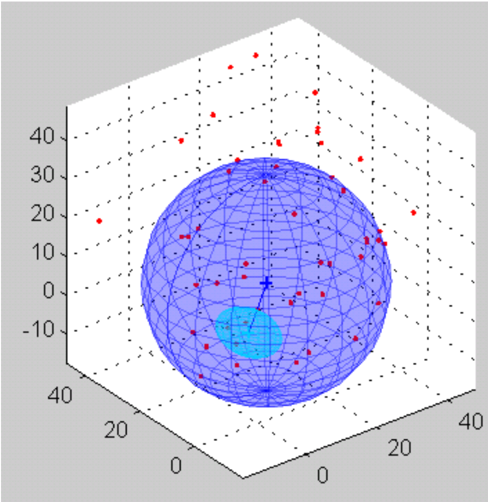


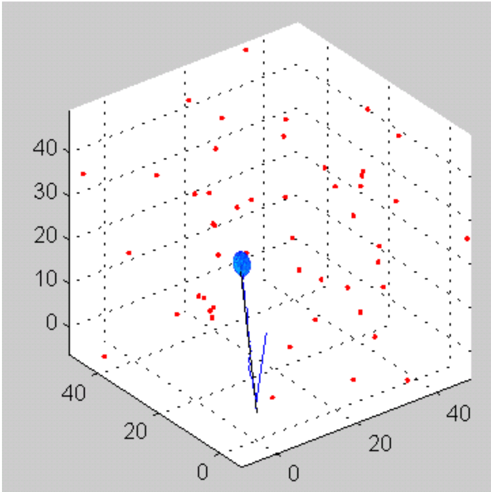
Fig 1. Result of position tracking. Black line: ground truth, blue line: estimated path, blue ellipsoid: 2σ bound of the covariance matrix. The notation is the same for the following plots.

In the formulation, the EKF algorithm can deal with small motion and measurement noises. Therefore, the algorithm can handle position tracking problem with reasonable accuracy. However, for global localization, we can set the covariance matrix Σ large enough to cover the possible location where the robot might be. In the following experiment, $\Sigma = \begin{bmatrix} 1000 & 0 & 0 \\ 0 & 1000 & 0 \\ 0 & 0 & 1000 \end{bmatrix}$, $\mu = 10$, $Q = 100^2$. Experimental result in figure 2 shows that the covariance converges to the correct location. In sum, the above

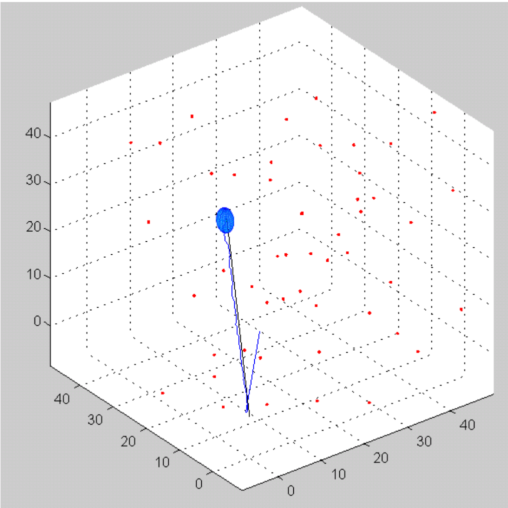
EKF formulation is can be applied to global localization is we set the initial covariance large enough to cover the real position.



(a) T=0



(b) T≈50



(c) T=100

Fig 2. Global localization experimental result.

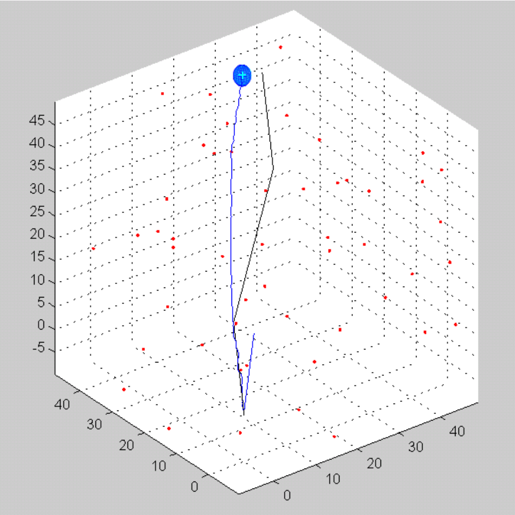


Fig 3. The failure case for kidnapped robot problem. The robot (black line) was arbitrarily moved to (30,30,30) at T=50.

For kidnapped robot problem, the robot is randomly moved to another place. To solve this problem, the robot should be able to tell whether the covariance ellipsoidal covers the ground truth by examining the difference between data and the expected measurement. Then, utilize a global localization solution to locate its current location. Afterward, the EKF position tracking can function normally.