Sheet 2
Motion Models and Robot Odometry

Group 4
Urs Borrmann, Caner Hazirbas, FangYi Zhi

May 22, 2013

Exercise 1

c) Group Picture

Figure 1: Group Photo

d) Graph Visualization

Figure 2 is the screenshot taken on rxgraph, shows the running nodes and topics.

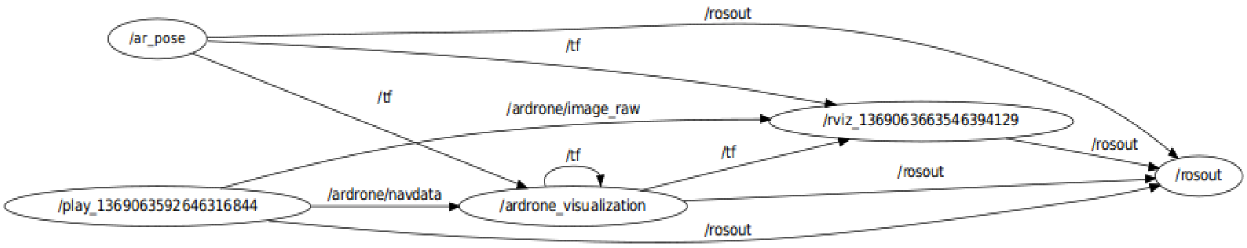

Figure 2: Running nodes and published topics

Exercise 2

d) Kalman Filter Covariance Ellipse Screenshot

Figure 3 is the screenshot of the covariance ellipse from the original Q matrix visualized by rviz.[1]

Figure 4 shows the screenshot of the estimated two-dimensional trajectory from the given bag file.

e) Kalman Filter with Higher Noise Screenshots

Figure 5 shows the screenshot of the covariance ellipse from the modified Q matrix, which drifts two times more in the global x-direction.

Figure 6 shows the screenshot of the estimated two-dimensional trajectory from the bag file with modified Q matrix.[2]

f) Noise Prediction for Experimental Setup

One can fly the quadracopter proportional to the marker located on the ground and read the values from sensors. After a while without changing position or

[1] In order to easily compare the difference between the effect of the two diffenrent Q matrices, we took both screenshots at the last moment of the given bagfile.

[2] There is no difference of the trajectories between the two different Q matrix, since until now, the Q matrix just adjusts the covariance ellipse but not the state vector, hence the trajectory.

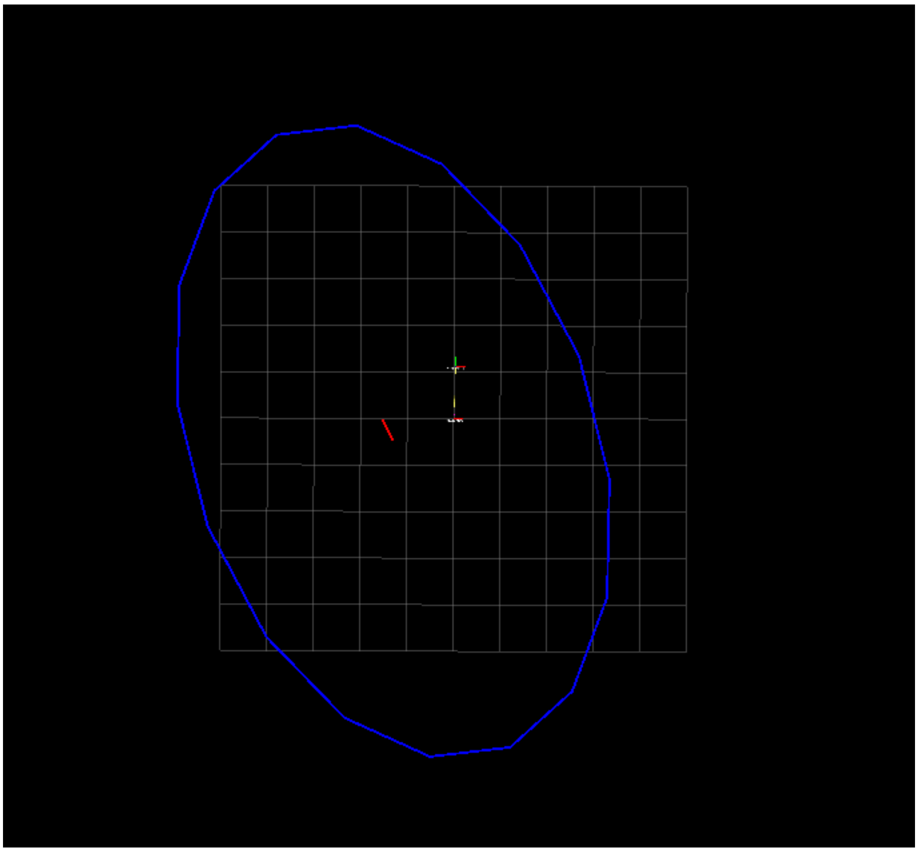

Figure 3: Covariance ellipse with the original Q matrix

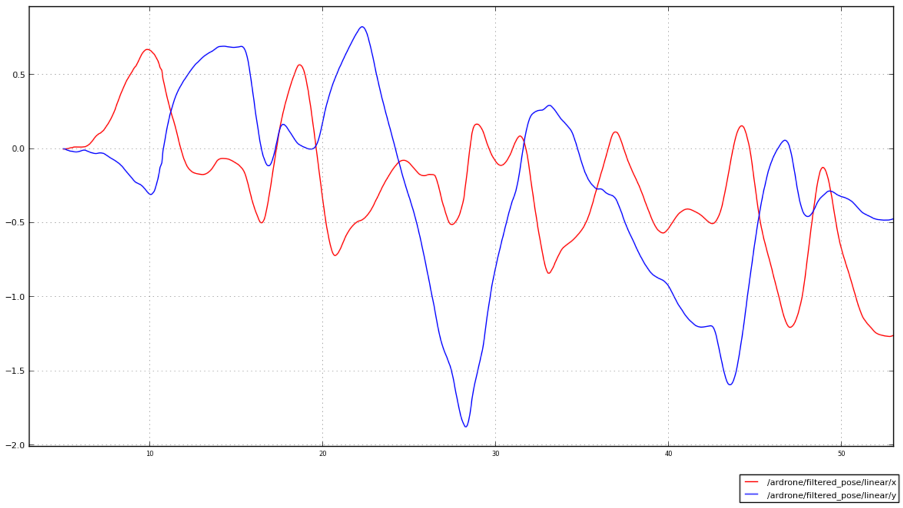

Figure 4: Estimated two-dimensional trajectory from the given bag file

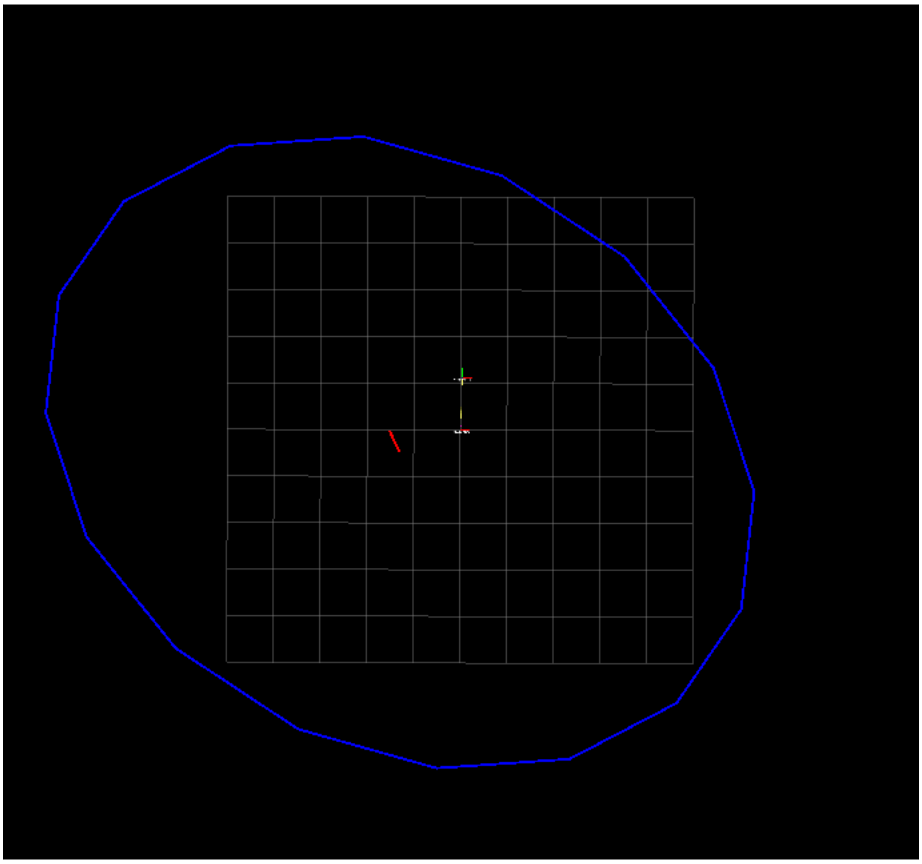

Figure 5: Covariance ellipse with the modified Q matrix

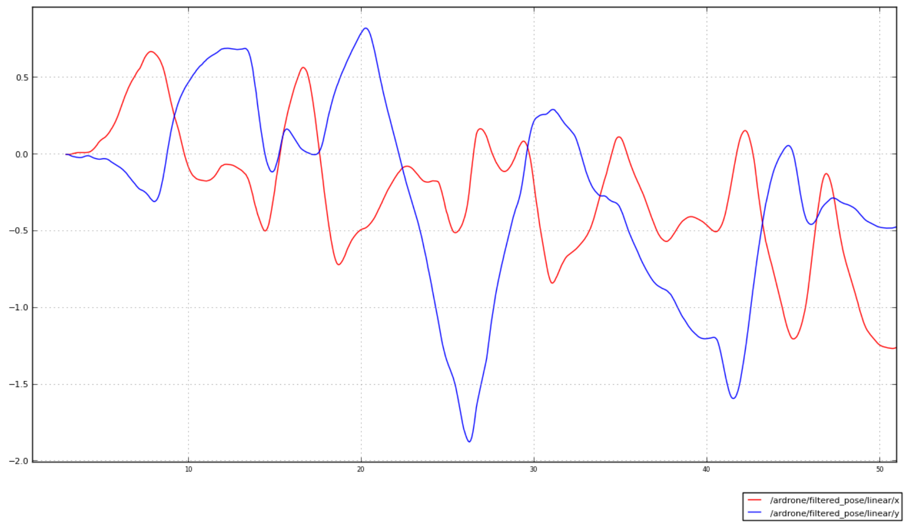

Figure 6: Estimated two-dimensional trajectory from the bag file with modified Q matrix.

orientation of quadracopter, we can observe how much drift we are having. Using observed amount of drift and waited time, one can estimate the noise.

g) Observation Function and Its Jacobian

Observation Function

Observed marker pose is calculated with function $h(x)$ (eq. 1). This $h(x)$ observation function predicts the marker pose z_{pre} given x, estimated robot world state(eq 2), and z_g ,the marker pose in global frame(eq. 3).

1. $$z_{pre} = h(x)$$ $$z_{pre} = (x_{pre} \quad y_{pre} \quad \psi_{pre})^T$$
2. $$x = (x_w \quad y_w \quad \psi_w)^T$$
3. $$z_g = (x_g \quad y_g \quad \psi_g)^T$$

In order to find the observation, we need to transform the global marker pose to local frame.

If X is homogeneous transformation matrix of x, robot pose,

$$X = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos\psi_w & -\sin\psi_w & x_w \\ \sin\psi_w & \cos\psi_w & y_w \\ 0 & 0 & 1 \end{pmatrix}$$

then we can transform any local frame to global frame as follows;

$$\vec{t_g} = X\vec{t_{pre}}$$

We want to transform from global to local. In order to do that we should take inverse of X transformation matrix;

$$X^{-1} = \begin{pmatrix} R^{-1} & -R^{-1}t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos\psi_w & \sin\psi_w & -x_w\cos\psi_w - y_w\sin\psi_w \\ -\sin\psi_w & \cos\psi_w & x_w\sin\psi_w - y_w\cos\psi_w \\ 0 & 0 & 1 \end{pmatrix}$$

Now we can compute the local marker position from global marker position;

$$\vec{t_g} = \begin{pmatrix} x_g \\ y_g \end{pmatrix}$$

$$\tilde{t_{pre}} = X^{-1}\tilde{t_g} = \begin{pmatrix} (x_g - x_w)\cos\psi_w + (y_g - y_w)\sin\psi_w \\ -(x_g - x_w)\sin\psi_w + (y_g - y_w)\cos\psi_w \\ 1 \end{pmatrix}$$

Since yaw angle is always in the global frame, observed yaw angle is

$$\psi_{pre} = (\psi_w - \psi_g)$$

At the end we get following observation function

$$h(x) = \begin{pmatrix} (x_g - x_w)\cos\psi_w + (y_g - y_w)\sin\psi_w \\ -(x_g - x_w)\sin\psi_w + (y_g - y_w)\cos\psi_w \\ (\psi_w - \psi_g) \end{pmatrix}$$

Jacobian of Observation Function

Now we can compute the jacobian of observation function as following;

$$H = \frac{\partial h(x)}{\partial x} = \begin{pmatrix} \frac{\partial h(x)}{\partial x_w} & \frac{\partial h(x)}{\partial y_w} & \frac{\partial h(x)}{\partial \psi_w} \end{pmatrix}$$

$$= \begin{pmatrix} -\cos\psi_w & -\sin\psi_w & -(x_g - x_w)\sin\psi_w + (y_g - y_w)\cos\psi_w \\ \sin\psi_w & -\cos\psi_w & -(x_g - x_w)\cos\psi_w - (y_g - y_w)\sin\psi_w \\ 0 & 0 & -1 \end{pmatrix}$$

i) Trajectory

Figure 7 shows the screenshot of the EKF corrected two-dimensional trajectory from the given bag file.

j) Drift on Pose Estimation

At the end of the 48th second, roughly we have 1.5m drift in x direction and 1m in y direction.

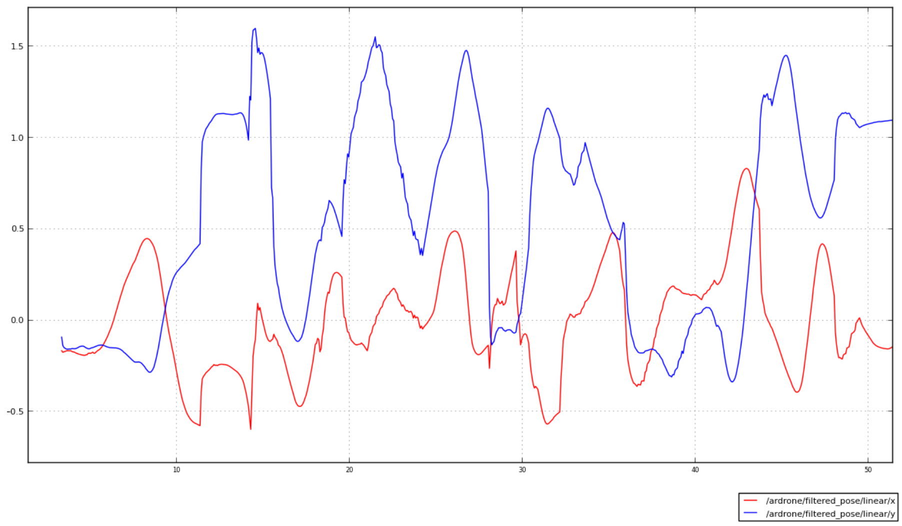

Figure 7: From the EKF corrected two-dimensional trajectory from the given bag file