SLAM and factorgraphs

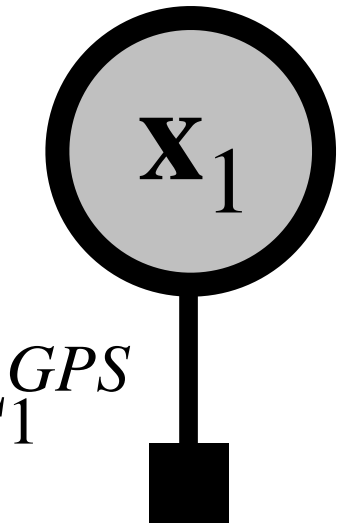
Labs 03

Karel Zimmermann

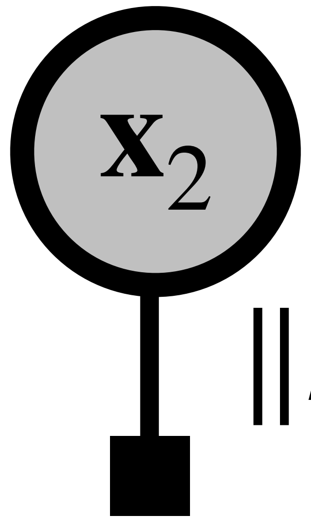
$$p\Bigg(\underbrace{\begin{bmatrix} z_t^{\text{GPS},x} \\ z_t^{\text{GPS},y} \end{bmatrix}}_{\mathbf{z}_t^{\text{GPS}}} \mid \underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{\mathbf{x}_t} \Bigg) = \mathcal{N}\Bigg(\mathbf{z}_t^{\text{GPS}}; \quad \underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{h^{\text{GPS}}(\mathbf{x}_t)} \quad , \ \mathbf{Q}_t^{\text{GPS}} \Bigg)$$

GPS

what is -log p(z|x)?



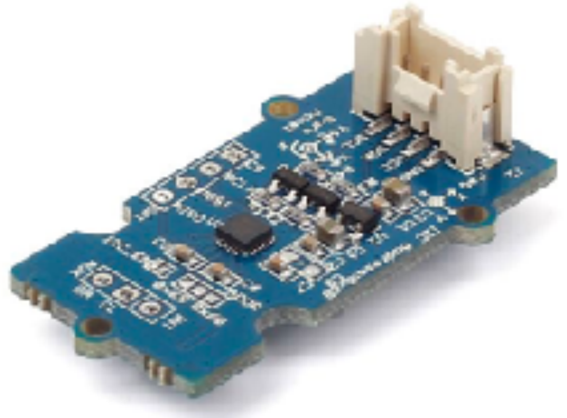
$$\|h^{\text{GPS}}(\mathbf{x}_1) - \mathbf{z}_1^{\text{GPS}}\|^2_{\Sigma_1^{GPS}}$$



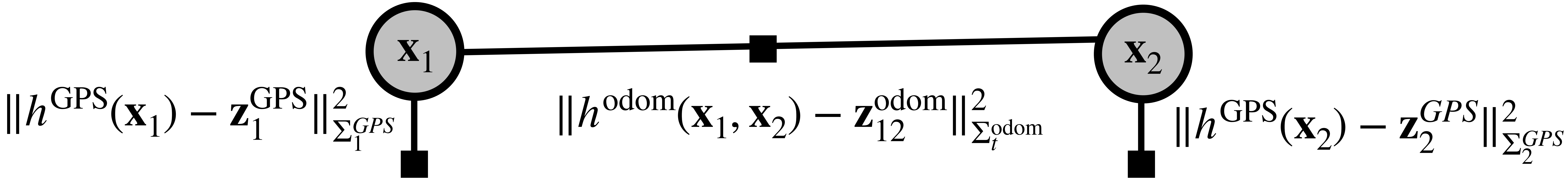
$$\|h^{\text{GPS}}(\mathbf{x}_2) - \mathbf{z}_2^{GPS}\|^2_{\Sigma_2^{GPS}}$$

$$p\left(\underbrace{\begin{bmatrix} z_t^{\text{GPS},x} \\ z_t^{\text{GPS},y} \end{bmatrix}}_{\mathbf{z}_t^{\text{GPS}}} \mid \underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{\mathbf{x}_t} \right) = \mathcal{N}\left(\mathbf{z}_t^{\text{GPS}};\ \underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{h^{\text{GPS}}(\mathbf{x}_t)},\ \mathbf{Q}_t^{\text{GPS}} \right)$$

$$p\left(\underbrace{\begin{bmatrix} z_t^{x} \\ z_t^{y} \end{bmatrix}}_{\mathbf{z}_t^{\text{odom}}} \mid \underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{\mathbf{x}_t}, \underbrace{\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix}}_{\mathbf{x}_{t+1}} \right) = \mathcal{N}\left(\mathbf{z}_t^{\text{odom}};\ \underbrace{\mathbf{x}_{t+1} - \mathbf{x}_t}_{h^{\text{odom}}(\mathbf{x}_t, \mathbf{x}_{t+1})},\ \mathbf{Q}_t^{\text{odom}} \right)$$

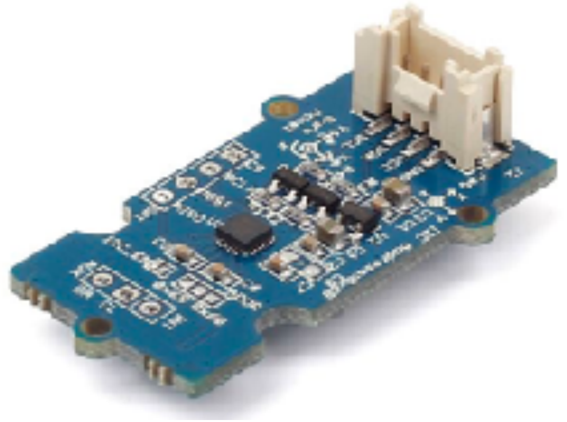


what is -log p(z_odom | x1, x2)?

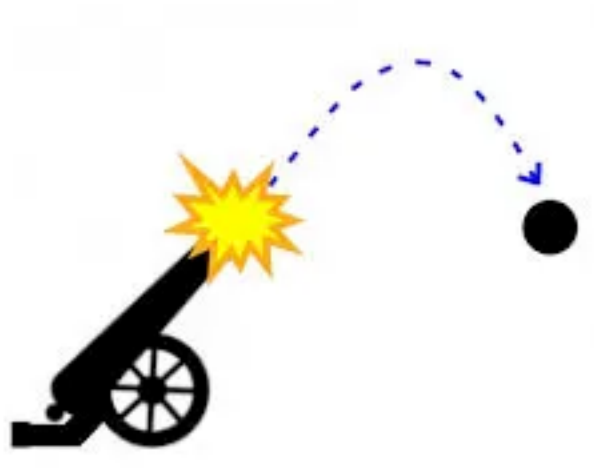


$$p\left(\underbrace{\begin{bmatrix} z_t^{\text{GPS},x} \\ z_t^{\text{GPS},y}\end{bmatrix}}_{\mathbf{z}_t^{\text{GPS}}} \mid \underbrace{\begin{bmatrix} x_t \\ y_t\end{bmatrix}}_{\mathbf{x}_t}\right) = \mathcal{N}\left(\mathbf{z}_t^{\text{GPS}}; \underbrace{\begin{bmatrix} x_t \\ y_t\end{bmatrix}}_{h^{\text{GPS}}(\mathbf{x}_t)}, \mathbf{Q}_t^{\text{GPS}}\right)$$

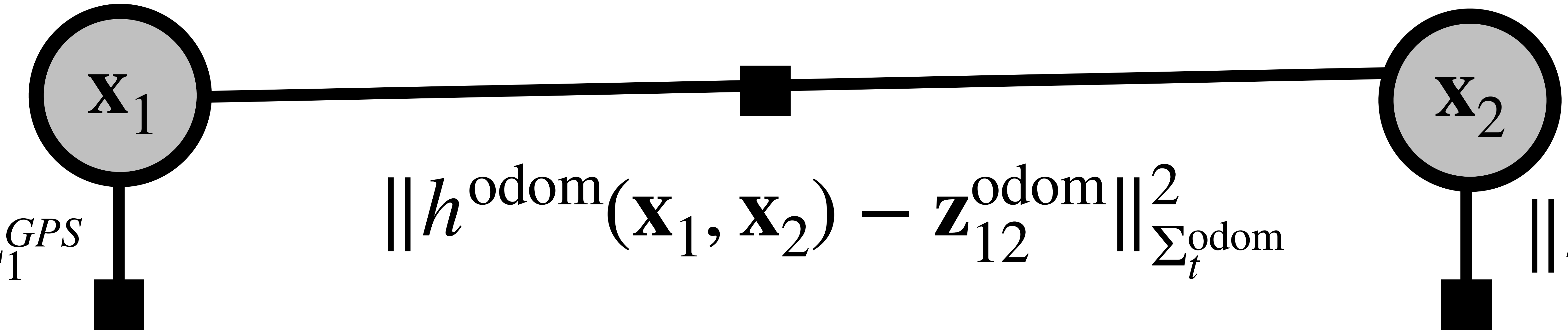
$$p\left(\underbrace{\begin{bmatrix} z_t^{x} \\ z_t^{y}\end{bmatrix}}_{\mathbf{z}_t^{\text{odom}}} \mid \underbrace{\begin{bmatrix} x_t \\ y_t\end{bmatrix}}_{\mathbf{x}_t}, \underbrace{\begin{bmatrix} x_{t+1} \\ y_{t+1}\end{bmatrix}}_{\mathbf{x}_{t+1}}\right) = \mathcal{N}\left(\mathbf{z}_t^{\text{odom}}; \underbrace{\mathbf{x}_{t+1} - \mathbf{x}_t}_{h^{\text{odom}}(\mathbf{x}_t, \mathbf{x}_{t+1})}, \mathbf{Q}_t^{\text{odom}}\right)$$



$$p\left(\underbrace{\begin{bmatrix} x_t \\ y_t\end{bmatrix}}_{\mathbf{x}_t} \,\middle|\, \underbrace{\begin{bmatrix} x_{t-1} \\ y_{t-1}\end{bmatrix}}_{\mathbf{x}_{t-1}}, \underbrace{\begin{bmatrix} v_t \\ \omega_t\end{bmatrix}}_{\mathbf{u}_t}\right) = \mathcal{N}\left(\mathbf{x}_t; \underbrace{\begin{bmatrix} x_{t-1} + v_t \Delta t \cos(\omega_t) \\ y_{t-1} + v_t \Delta t \sin(\omega_t) - \frac{1}{2} g \Delta t^2\end{bmatrix}}_{g(\mathbf{x}_{t-1}, \mathbf{u}_t)}, \mathbf{R}_t\right)$$



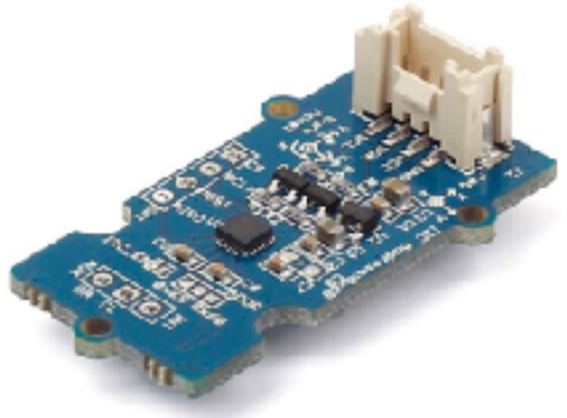
what is -log p(x2 | x1, u12)?



$$p\Bigg(\underbrace{\begin{bmatrix} z_t^{\text{GPS},x} \\ z_t^{\text{GPS},y} \end{bmatrix}}_{\mathbf{z}_t^{\text{GPS}}} \,\Bigg|\, \underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{\mathbf{x}_t}\Bigg) = \mathcal{N}\Bigg(\mathbf{z}_t^{\text{GPS}};\ \underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{h^{\text{GPS}}(\mathbf{x}_t)},\ \mathbf{Q}_t^{\text{GPS}}\Bigg)$$

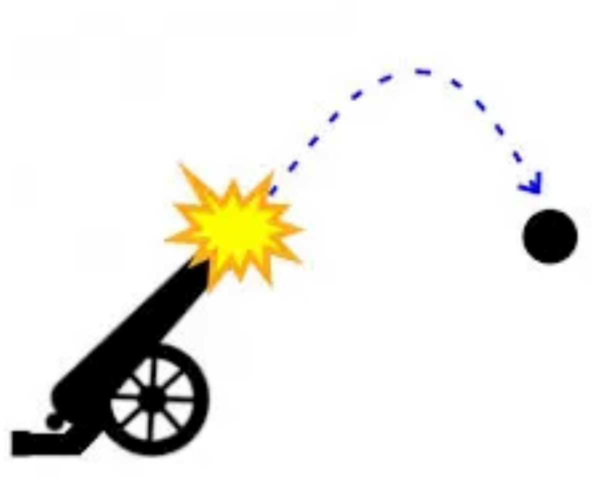
GPS

$$p\Bigg(\underbrace{\begin{bmatrix} z_t^{x} \\ z_t^{y} \end{bmatrix}}_{\mathbf{z}_t^{\text{odom}}} \,\Bigg|\, \underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{\mathbf{x}_t}, \underbrace{\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix}}_{\mathbf{x}_{t+1}}\Bigg) = \mathcal{N}\Bigg(\mathbf{z}_t^{\text{odom}};\ \underbrace{\mathbf{x}_{t+1} - \mathbf{x}_t}_{h^{\text{odom}}(\mathbf{x}_t, \mathbf{x}_{t+1})},\ \mathbf{Q}_t^{\text{odom}}\Bigg)$$



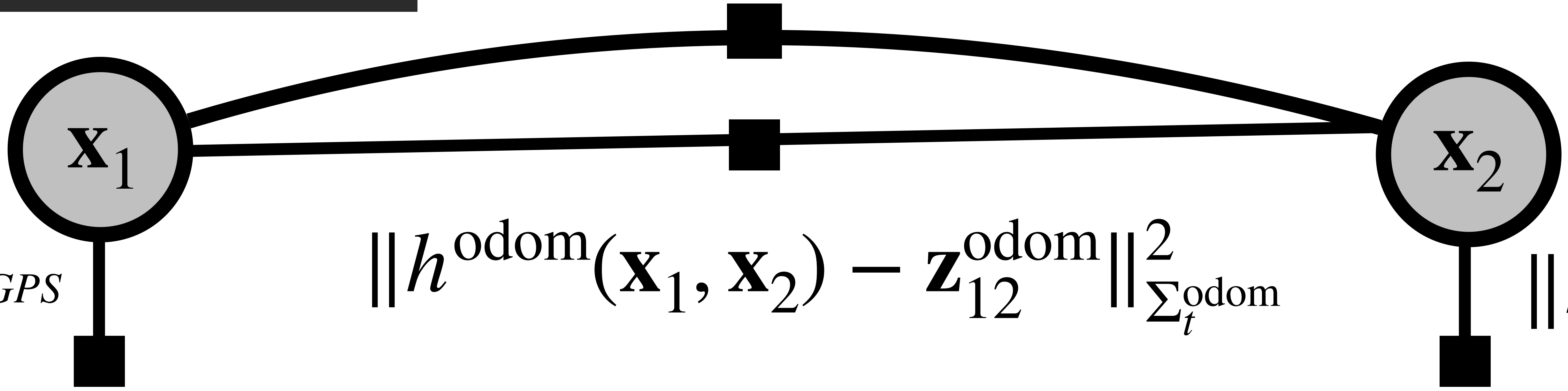
IMU

$$p\Bigg(\underbrace{\begin{bmatrix} x_t \\ y_t \end{bmatrix}}_{\mathbf{x}_t} \,\Bigg|\, \underbrace{\begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix}}_{\mathbf{x}_{t-1}}, \underbrace{\begin{bmatrix} v_t \\ \omega_t \end{bmatrix}}_{\mathbf{u}_t}\Bigg) = \mathcal{N}\Bigg(\mathbf{x}_t;\ \underbrace{\begin{bmatrix} x_{t-1} + v_t \Delta t \cos(\omega_t) \\ y_{t-1} + v_t \Delta t \sin(\omega_t) - \frac{1}{2} g \Delta t^2 \end{bmatrix}}_{g(\mathbf{x}_{t-1}, \mathbf{u}_t)},\ \mathbf{R}_t\Bigg)$$

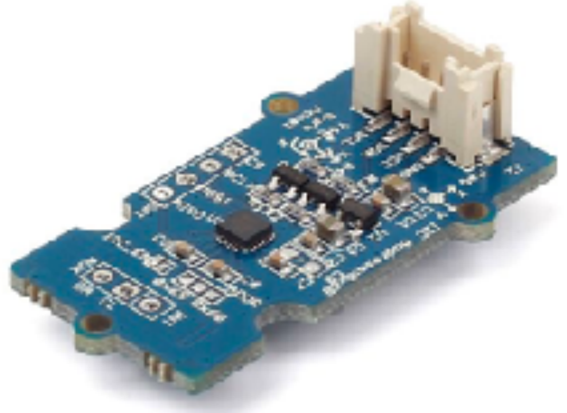


motion model

```
loss = ...... # <------- TODO FILL IN
```



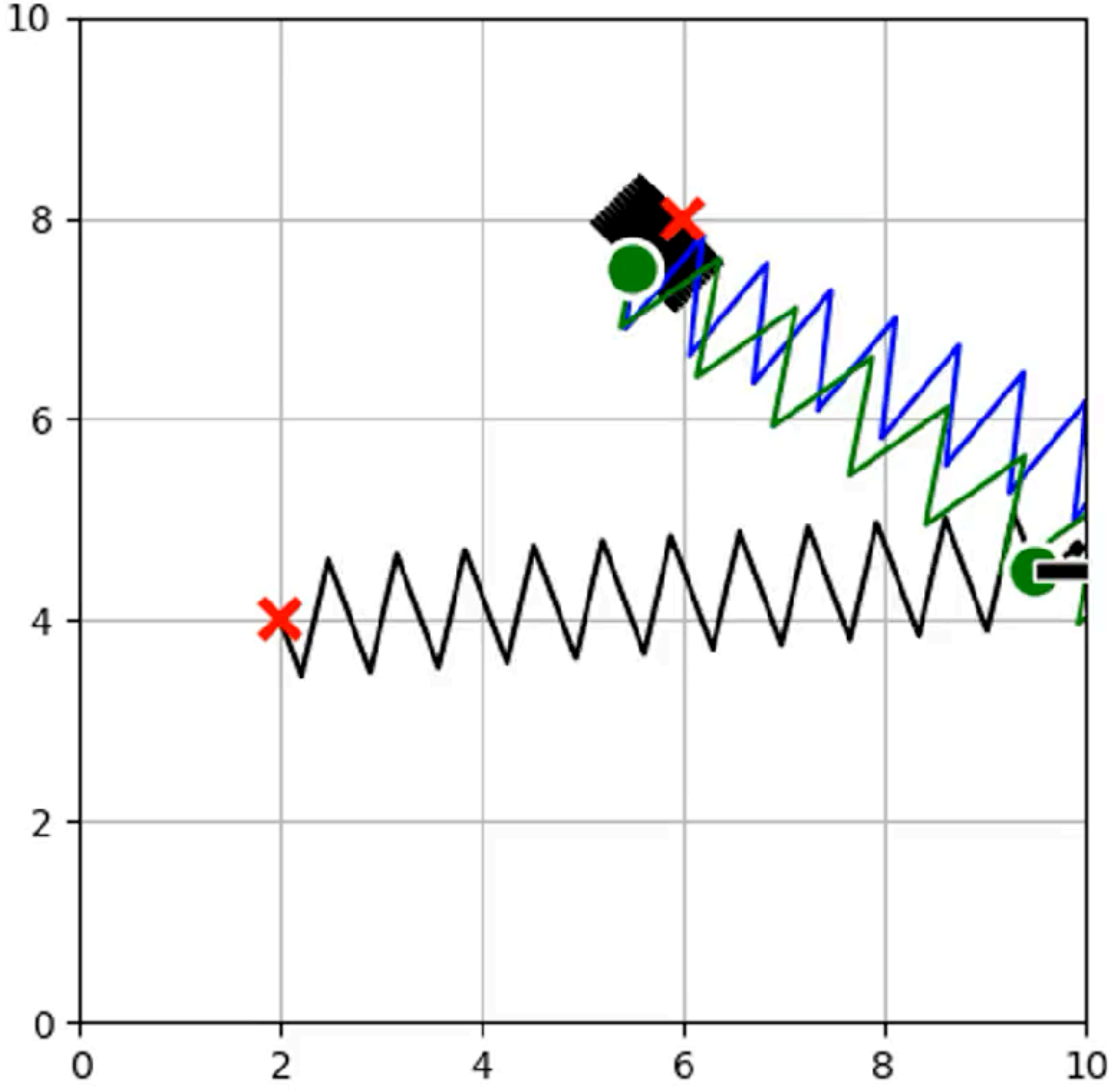
GPS +



IMU +

motion model

$$\mathbf{x}_1^\star, \mathbf{x}_2^\star = \arg\min_{\mathbf{x}_1, \mathbf{x}_2} \ \|\mathbf{x}_1 - \mathbf{z}_1^{GPS}\|^2_{\Sigma_1^{GPS}} + \|\mathbf{x}_2 - \mathbf{z}_2^{GPS}\|^2_{\Sigma_2^{GPS}} + \|\mathbf{x}_2 - \mathbf{x}_1 - \mathbf{z}_{12}^{odom}\|^2_{\Sigma_t^{odom}} + \|g(\mathbf{x}_1, \mathbf{u}_2) - \mathbf{x}_2\|^2_{\Sigma_t^g}$$



$\mathbf{x}_t$ …robot poses

$\mathbf{z}_t^{GPS}$ …GPS measurement

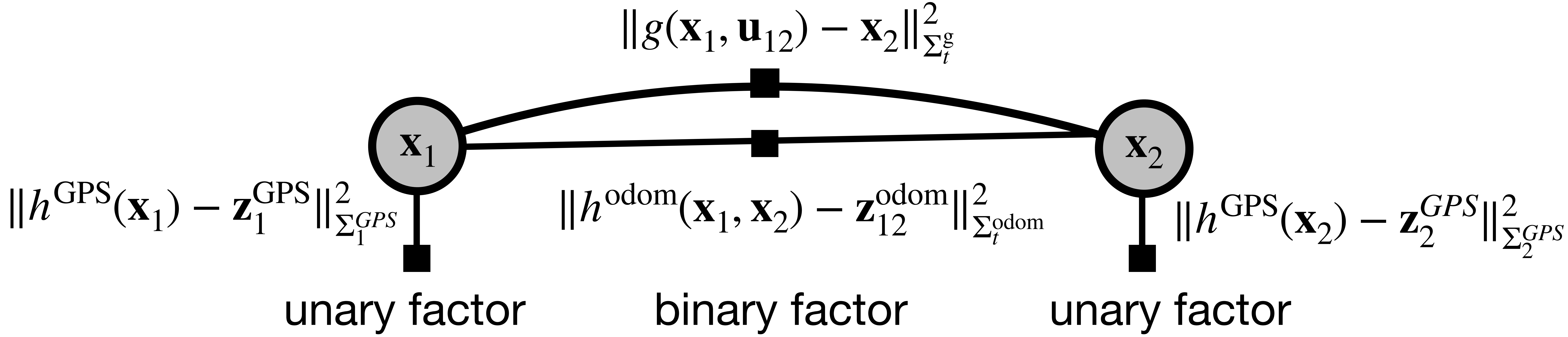
$\mathbf{z}_t^{odom}$ …odometry measurements

$\sum_t \|\mathbf{x}_t - \mathbf{z}_t^{GPS}\|^2_{\Sigma_t^{GPS}}$ … GPS loss

$\|\mathbf{x}_2 + \mathbf{z}_{12}^{odom} - \mathbf{x}_2\|^2_{\Sigma_t^{odom}}$ … odom loss

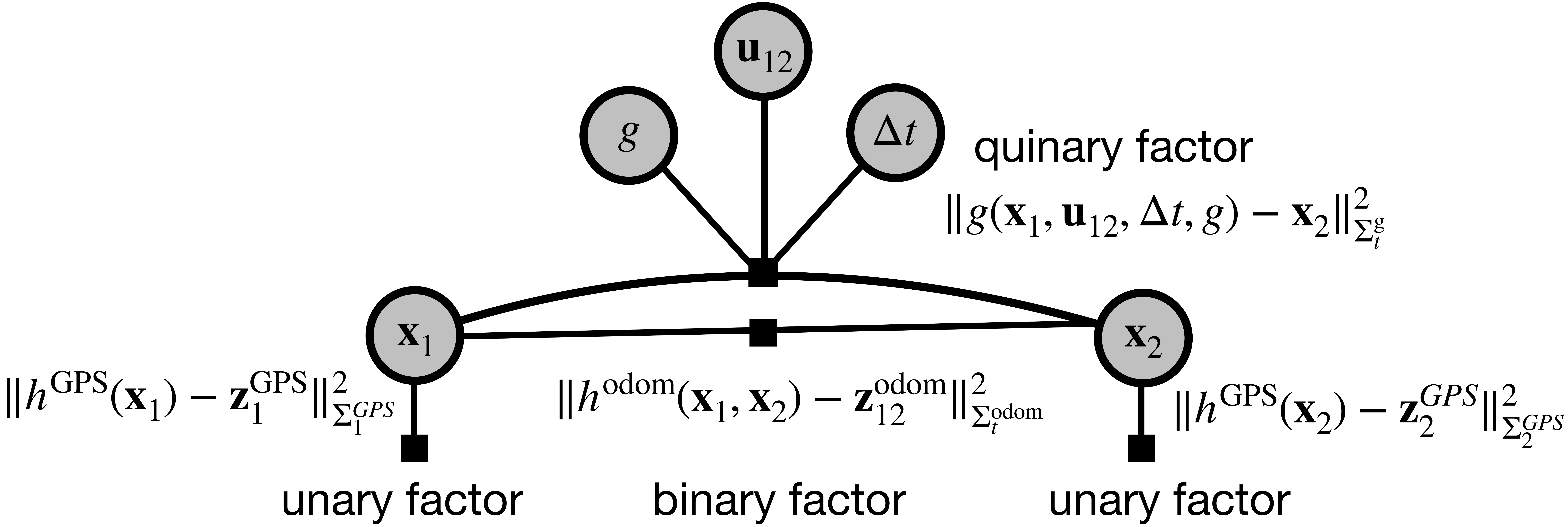
$\|g(\mathbf{x}_1, \mathbf{u}_2) - \mathbf{x}_2\|^2_{\Sigma_t^g}$ … motion loss

What if the control, gravitaty and time are also unknown?

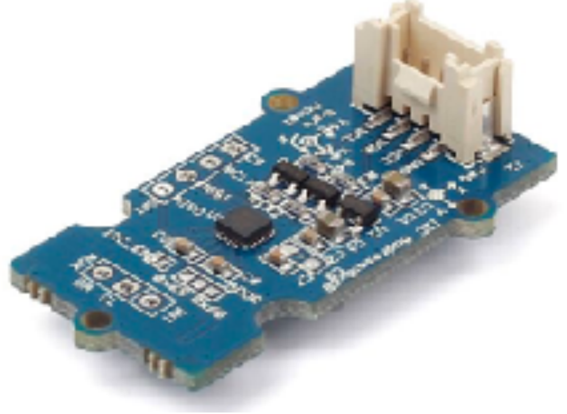


What if the control, gravitaty and time are also unknown?

```
opt = torch.optim.Adam([x1, x2, …… ], lr=0.5)
```



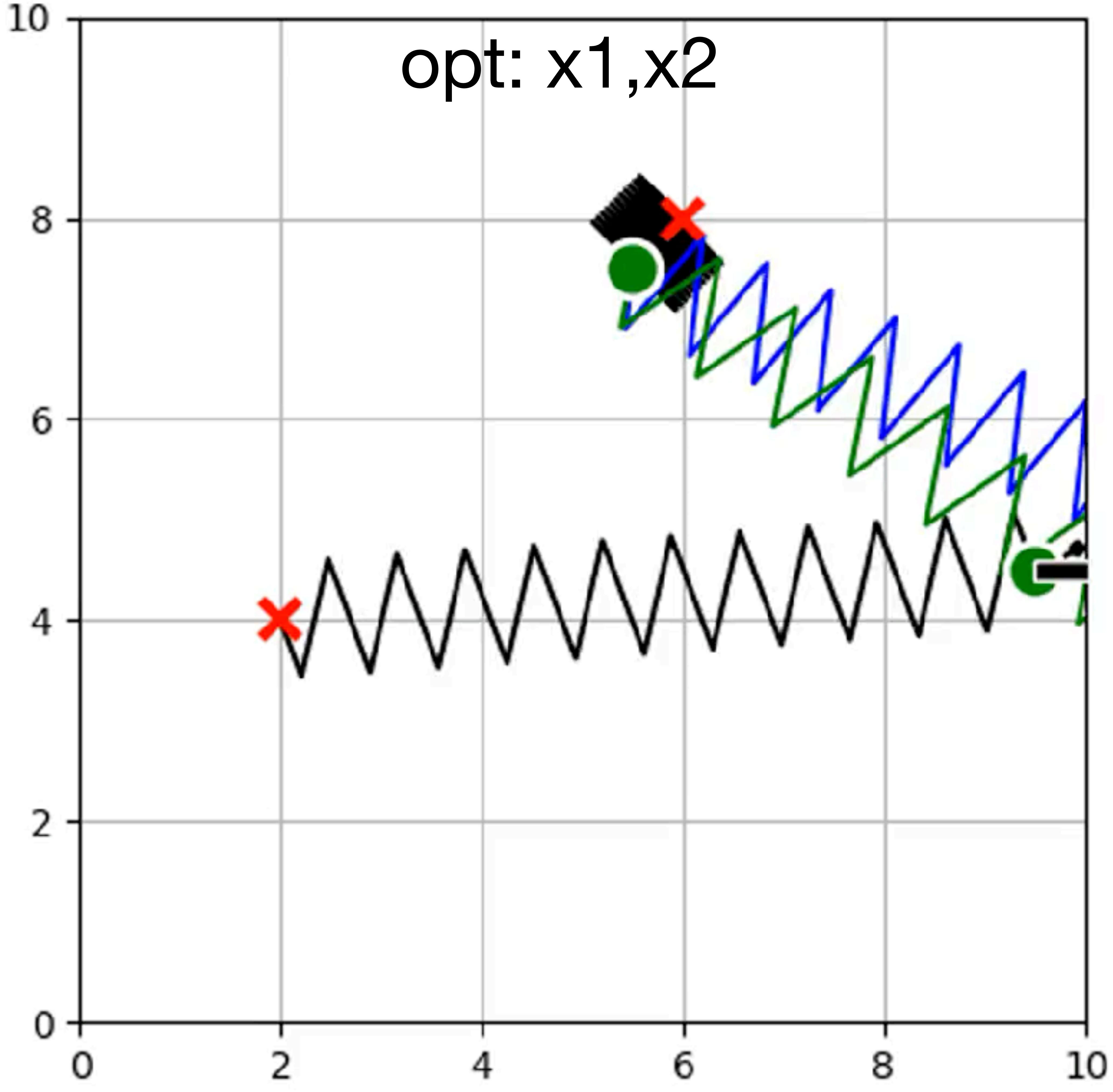
GPS +

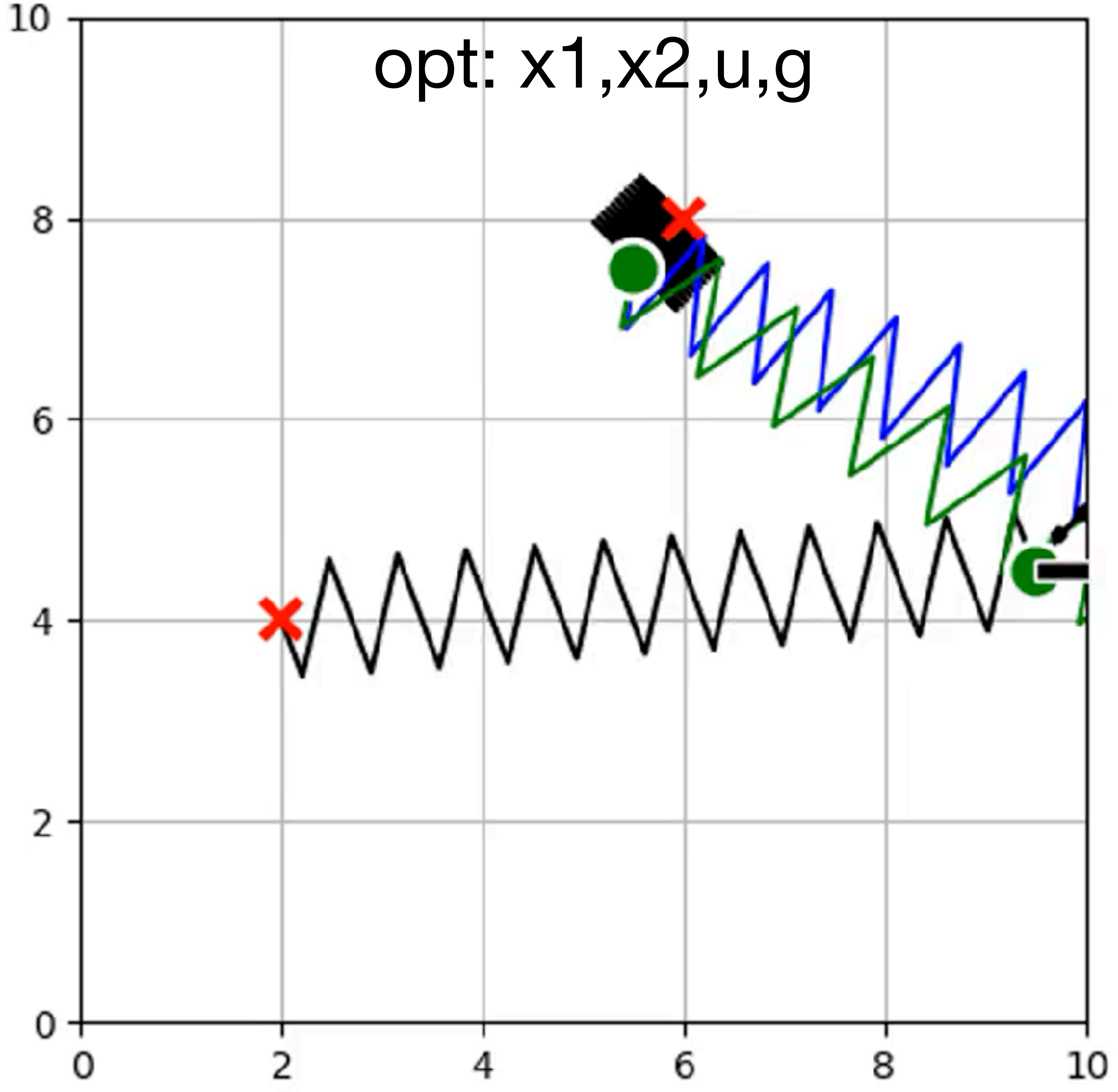


IMU +

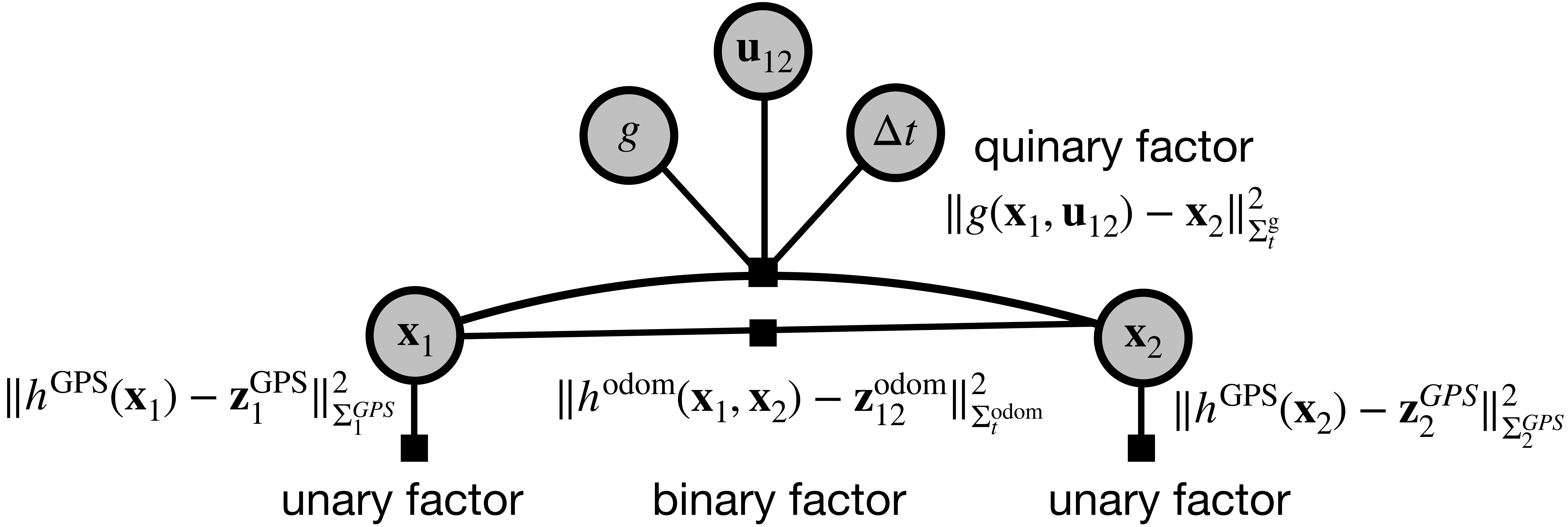
motion model

$$\mathbf{x}_1^\star, \mathbf{x}_2^\star = \arg\min_{\mathbf{x}_1,\mathbf{x}_2} \|\mathbf{x}_1 - \mathbf{z}_1^{GPS}\|^2_{\Sigma_1^{GPS}} + \|\mathbf{x}_2 - \mathbf{z}_2^{GPS}\|^2_{\Sigma_2^{GPS}} + \|\mathbf{x}_2 + \mathbf{z}_{12}^{odom} - \mathbf{x}_2\|^2_{\Sigma_t^{odom}} + \|g(\mathbf{x}_1, \mathbf{u}_2) - \mathbf{x}_2\|^2_{\Sigma_t^g}$$



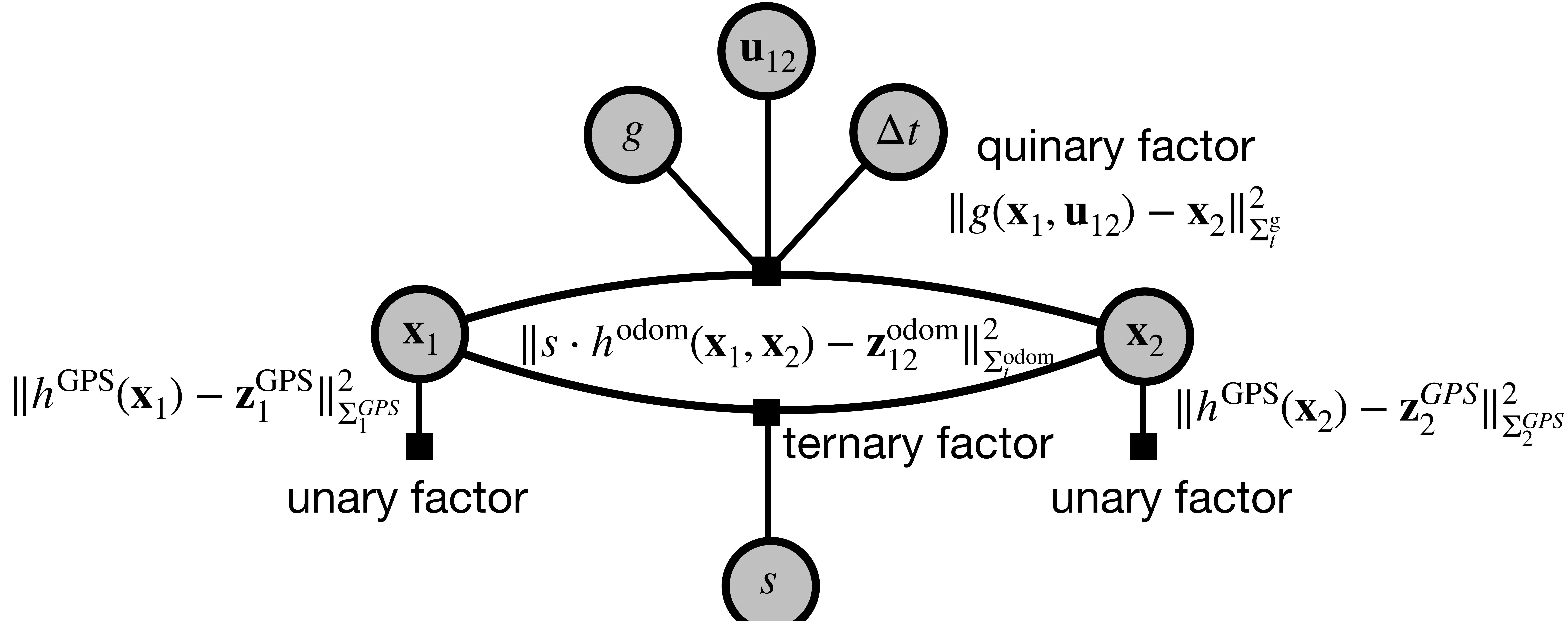


What if the scale of odometry measurement is unknown?



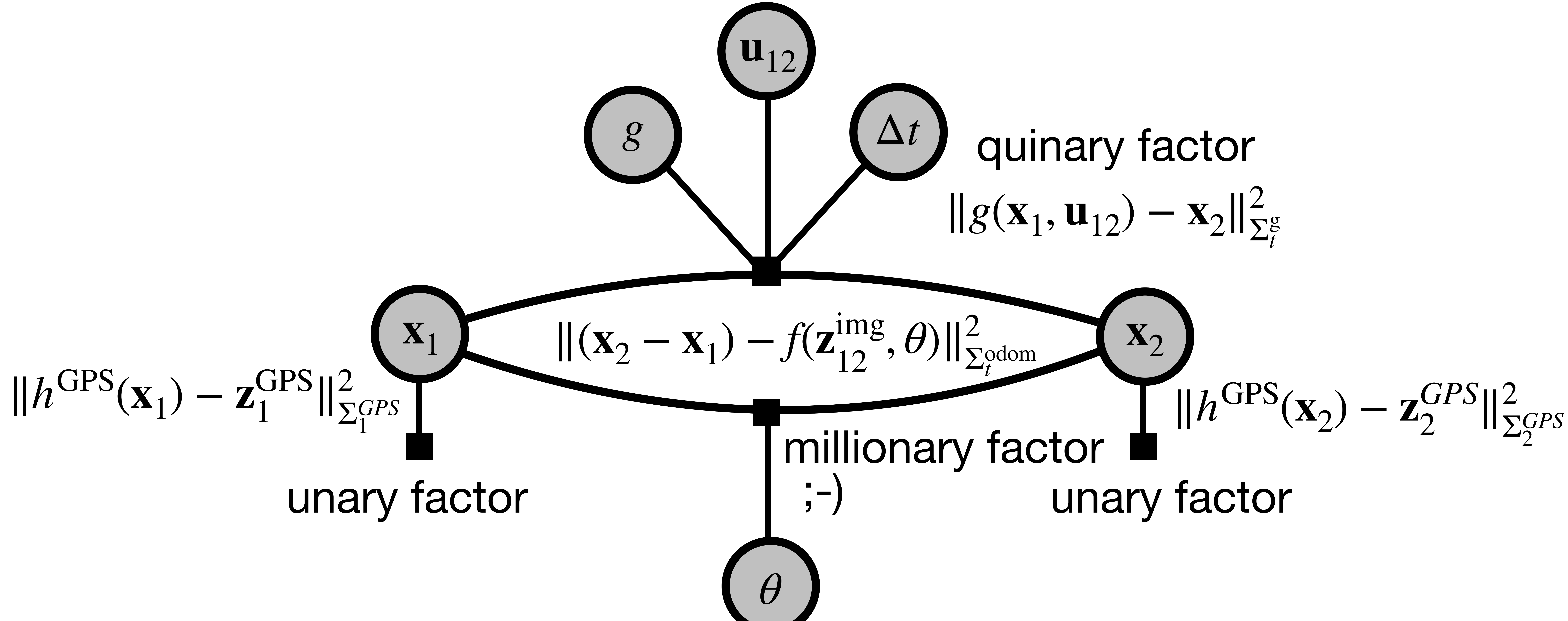
What if the scale of odometry measurement is unknown?

What if the measurement is blurred image and I want to learn deep odometry predictor, which transform image to odometry vector?

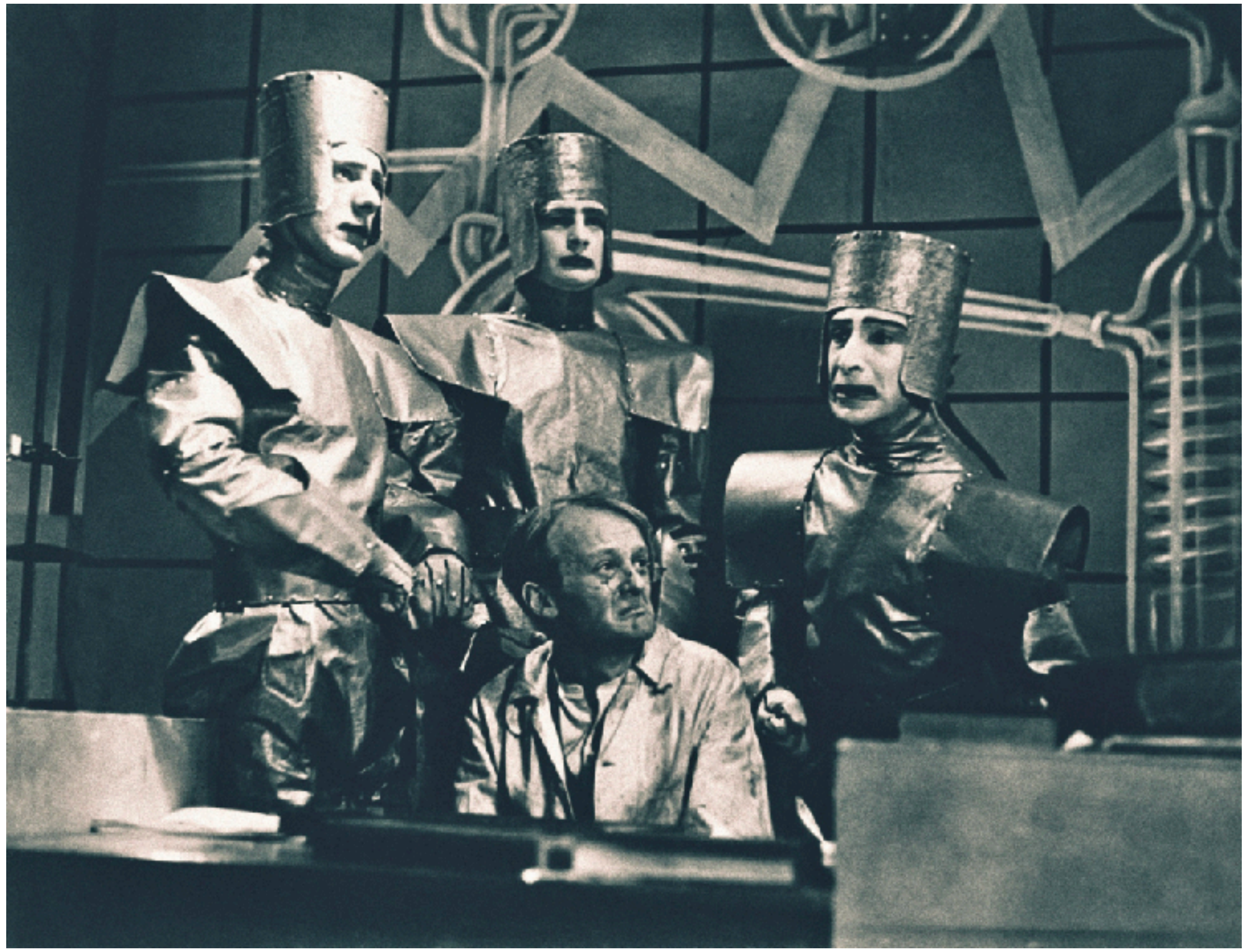


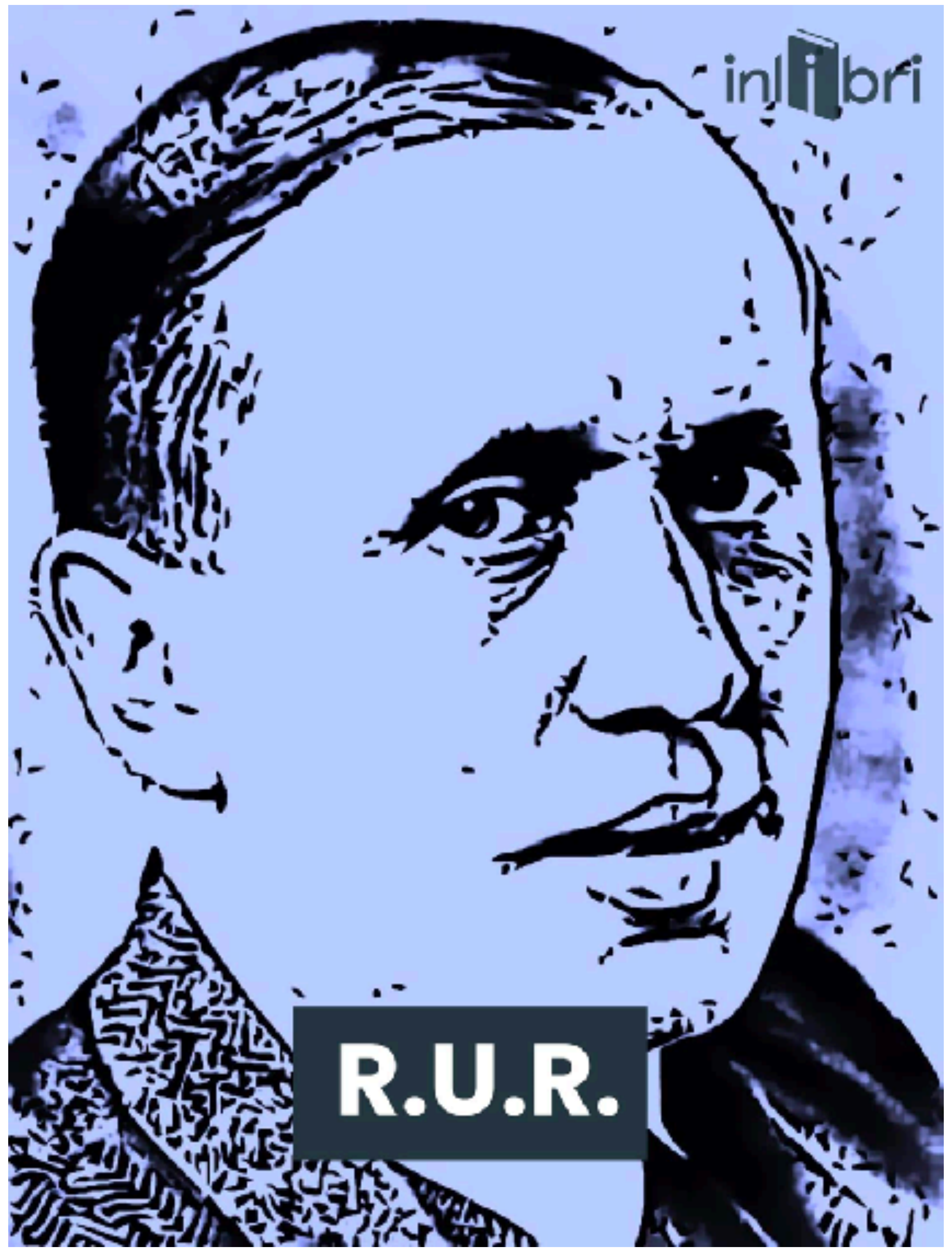
What if the scale of odometry measurement is unknown?

What if the measurement is blurred image and I want to learn deep odometry predictor, which transform image to odometry vector?




SVR → NN





Where does the image come from? Theatre play “Rossum's Universal Robots”1920

R.U.R. challenge: bonus task (max 3points)

- Come up with an original sci-fi postapocalyptic universe (image diffusion model)
- Unique law of physics (e.g. variable gravity even negative, speed of light = 60km/h)
- Unique sensor with non-linear measurement function (e.g. $z = \sin(x)$)
- Unique state estimation problem (describe + draw corresponding factorgraph)
- Implement two state example in pytorch
- Upload to BRUTE

Localization in SE(2)

Pose of the house transformed from rcf to wcf:

$$\mathbf{z}^w = \begin{bmatrix} z_x^w \\ z_y^w \\ z_\theta^w \end{bmatrix} = \begin{bmatrix} \cos\theta_t & -\sin\theta_t & 0 \\ \sin\theta_t & \cos\theta_t & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_x^r \\ z_y^r \\ z_\theta^r \end{bmatrix} + \begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix} = \begin{bmatrix} R(\theta_t) & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{bmatrix} \mathbf{z}^r + \mathbf{x}_t \; = T(\mathbf{z}^r, \mathbf{x}_t) \; \ldots \; \text{r2w}(\mathbf{z}^r, \mathbf{x}_t)$$

Pose of the house transformed from wcf to rcf:

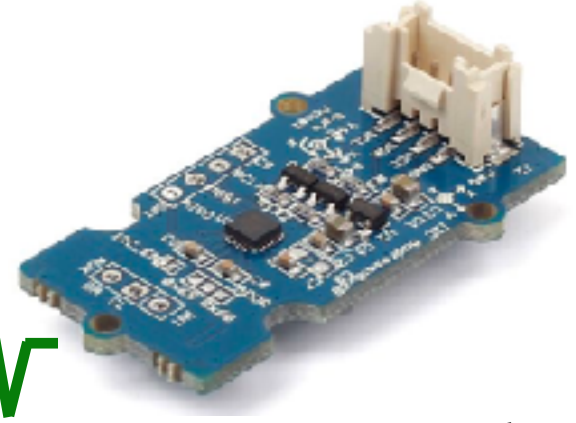
$$\mathbf{z}^r = \begin{bmatrix} R(\theta_t)^\top & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{bmatrix} (\mathbf{z}^w - \mathbf{x}_t) = T^{-1}(\mathbf{z}^w, \mathbf{x}_t) \quad \ldots \; \text{w2r}(\mathbf{z}^w, \mathbf{x}_t)$$

```
def r2w(z_r, x):
    z_w = torch.zeros(3)
    R = torch.vstack((torch.hstack((torch.cos(x[2]), -torch.sin(x[2]))), …
    z_w[0:2] = R @ z_r[0:2] + x[0:2]
    z_w[2] = z_r[2] + x[2]
    return z_w

def w2r(z_w, x):
    z_r = torch.zeros(3)
    R = torch.vstack((torch.hstack((torch.cos(x[2]), -torch.sin(x[2]))), …
    z_r[0:2] = R.t() @ (z_w[0:2] - x[0:2])
    z_r[2] = z_w[2] - x[2]
    return z_r
```

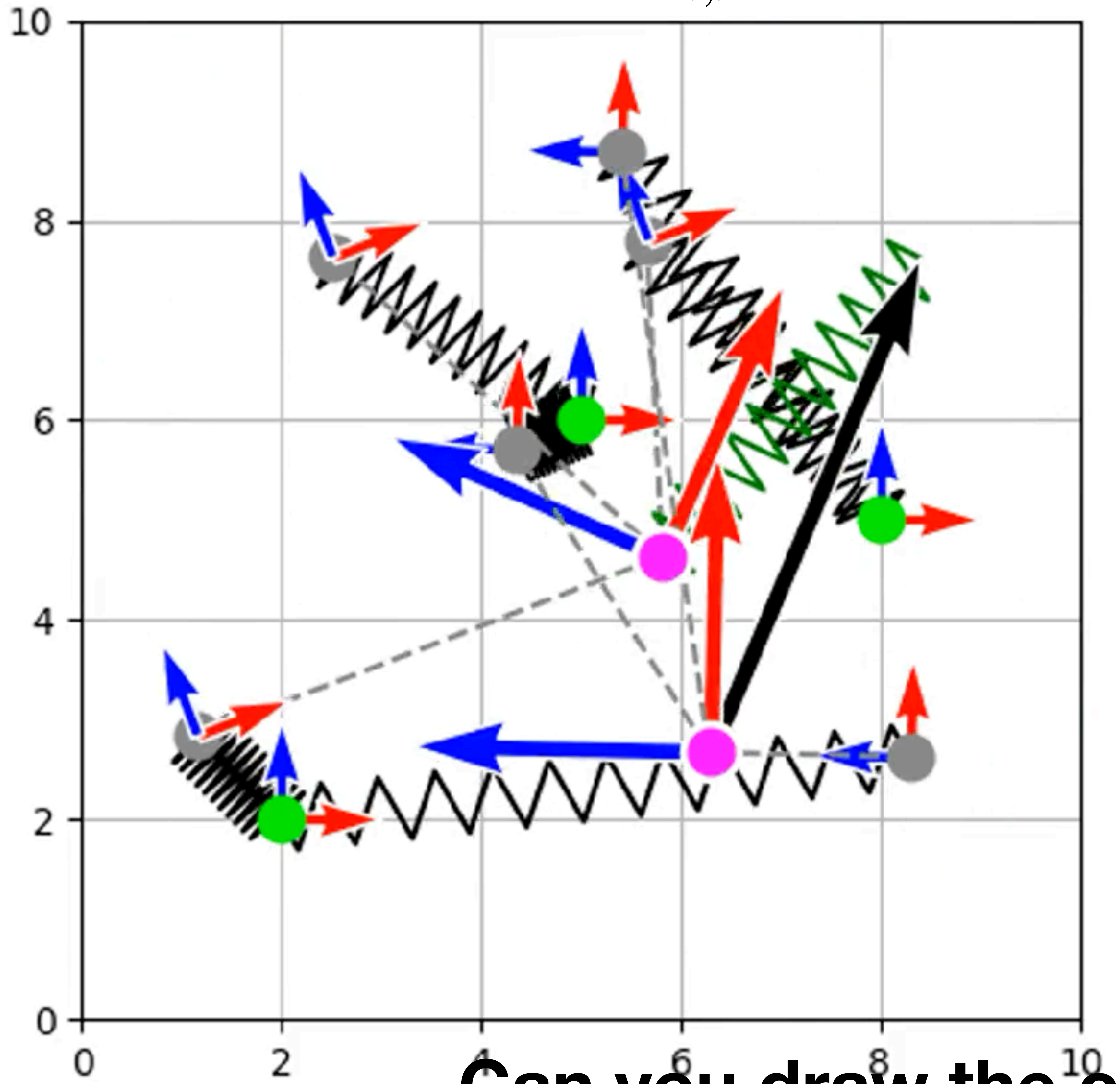
Localization

3D marker detector (RGBD camera) +



Odometry (IMU)

$$\mathbf{x}_1^\star, \mathbf{x}_2^\star = \arg\min_{\mathbf{x}_1,\mathbf{x}_2} \sum_{i,t} \|\text{w2r}(\mathbf{m}_i, \mathbf{x}_t) - \mathbf{z}_t^{\mathbf{m}_i}\|^2 + \|\text{w2r}(\mathbf{x}_2, \mathbf{x}_1) - \mathbf{z}_{12}^{odom}\|^2$$



$\mathbf{x}_t$ … robot poses

$\mathbf{m}_i$ … known marker positions

$\mathbf{z}_t^{\mathbf{m}_i}$ … marker measurements

…… local coordinate frame

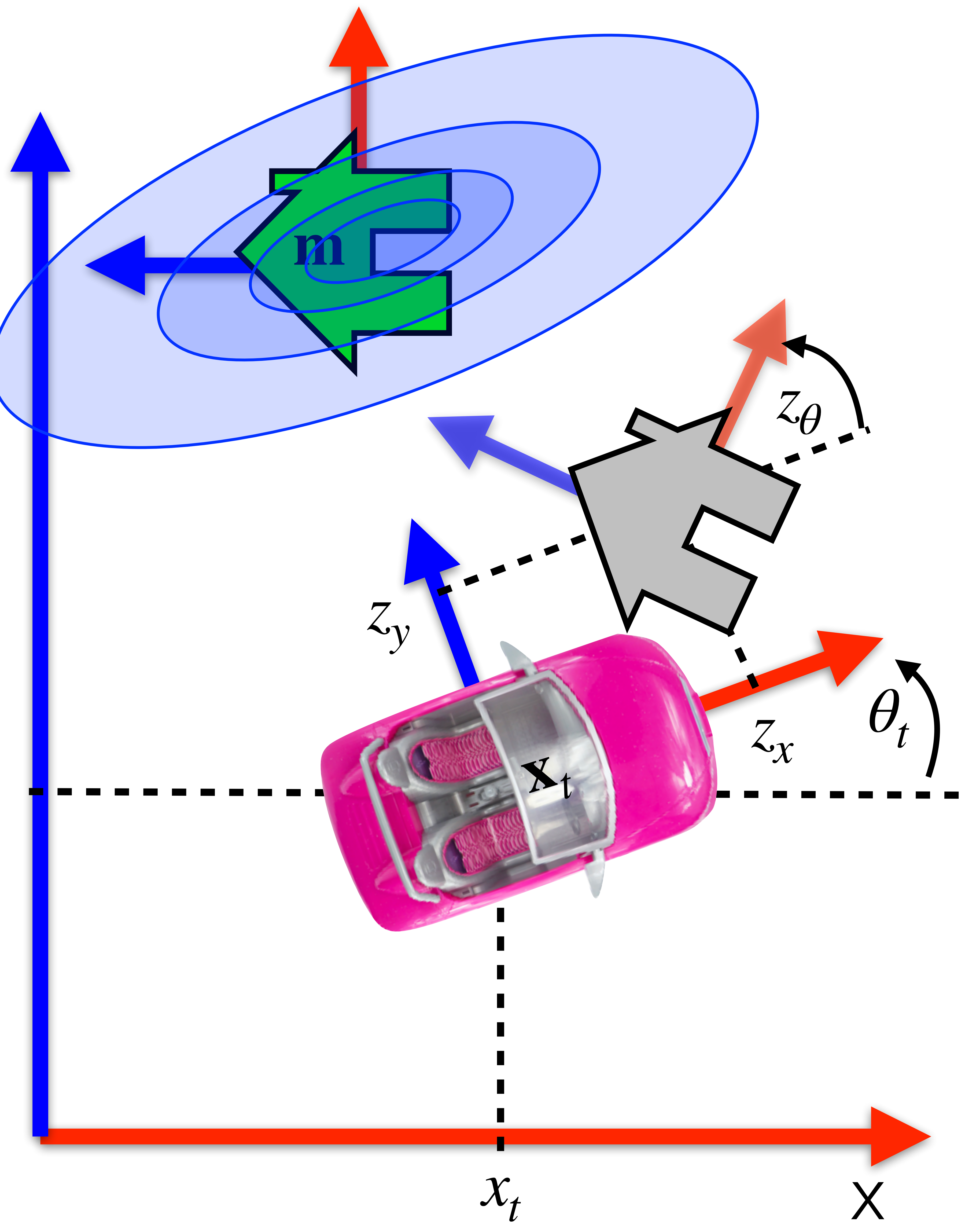
…… odometry

$\sum_{i,t} \|\text{w2r}(\mathbf{m}_i, \mathbf{x}_t) - \mathbf{z}_t^{\mathbf{m}_i}\|^2$ … marker loss

$\|\text{w2r}(\mathbf{x}_2, \mathbf{x}_1) - \mathbf{z}_{12}^{odom}\|^2$ … odom loss

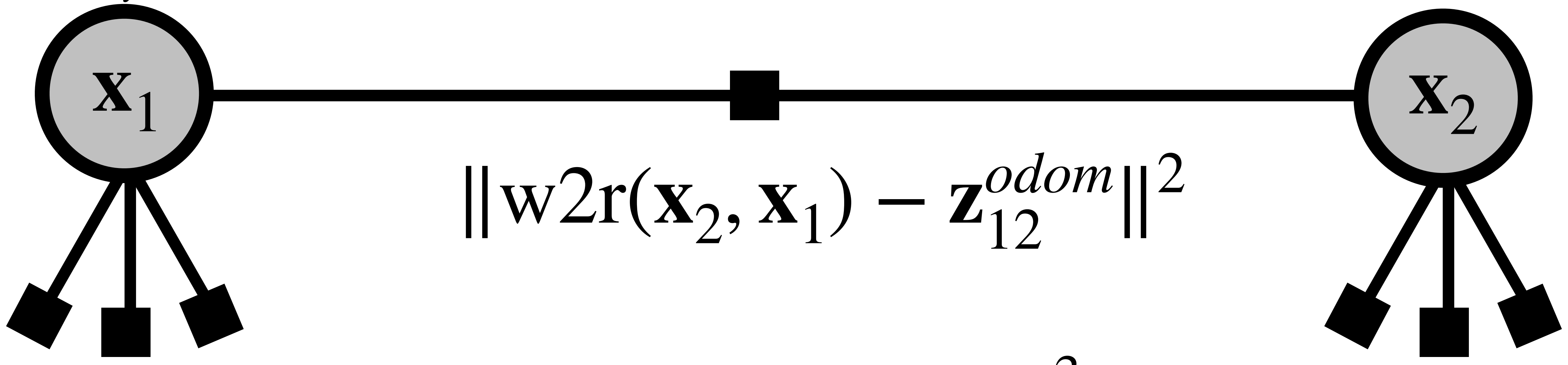
Can you draw the corresponding factorgraph?

Marker detector in EKF SLAM



$$p\left(\underbrace{\begin{bmatrix} z_t^{\mathbf{m},x} \\ z_t^{\mathbf{m},y} \\ z_t^{\mathbf{m},\theta} \end{bmatrix}}_{\mathbf{z}_t^{\mathbf{m}}} \;\middle|\; \underbrace{\begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix}}_{\mathbf{x}_t}\right) = \mathcal{N}\left(\mathbf{z}_t^{\mathbf{m}};\ \underbrace{\mathrm{w2r}(\mathbf{m}, \mathbf{x_t})}_{h^{\mathbf{m}}(\mathbf{x}_t)},\ Q_t^{\mathbf{m}}\right)$$

$$p\left(\underbrace{\begin{bmatrix} z_t^{\mathrm{odom},x} \\ z_t^{\mathrm{odom},y} \\ z_t^{\mathrm{odom},\theta} \end{bmatrix}}_{\mathbf{z}_t^{\mathrm{odom}}} \;\middle|\; \underbrace{\begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix}}_{\mathbf{x}_t}\right) = \mathcal{N}\left(\mathbf{z}_t^{\mathrm{odom}};\ \underbrace{\mathrm{w2r}(\mathbf{m}, \mathbf{x_t})}_{h^{\mathrm{odom}}(\mathbf{x}_t)},\ Q_t^{\mathrm{odom}}\right)$$



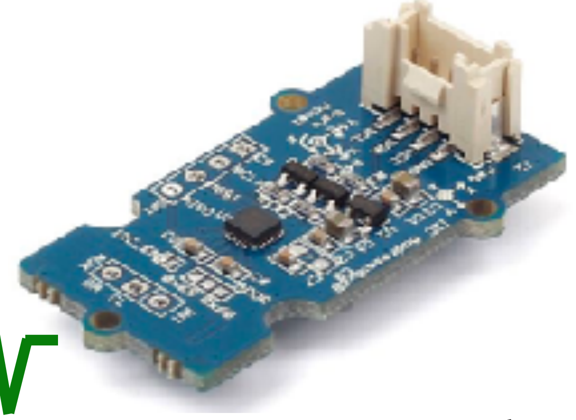
$$\|\mathrm{w2r}(\mathbf{x}_2, \mathbf{x}_1) - \mathbf{z}_{12}^{odom}\|^2$$

$$\sum_{i=1}^{3} \|\mathrm{w2r}(\mathbf{m}_i, \mathbf{x}_1) - \mathbf{z}_1^{\mathbf{m}_i}\|^2 \qquad \sum_{i=1}^{3} \|\mathrm{w2r}(\mathbf{m}_i, \mathbf{x}_2) - \mathbf{z}_2^{\mathbf{m}_i}\|^2$$

SLAM in SE(2)

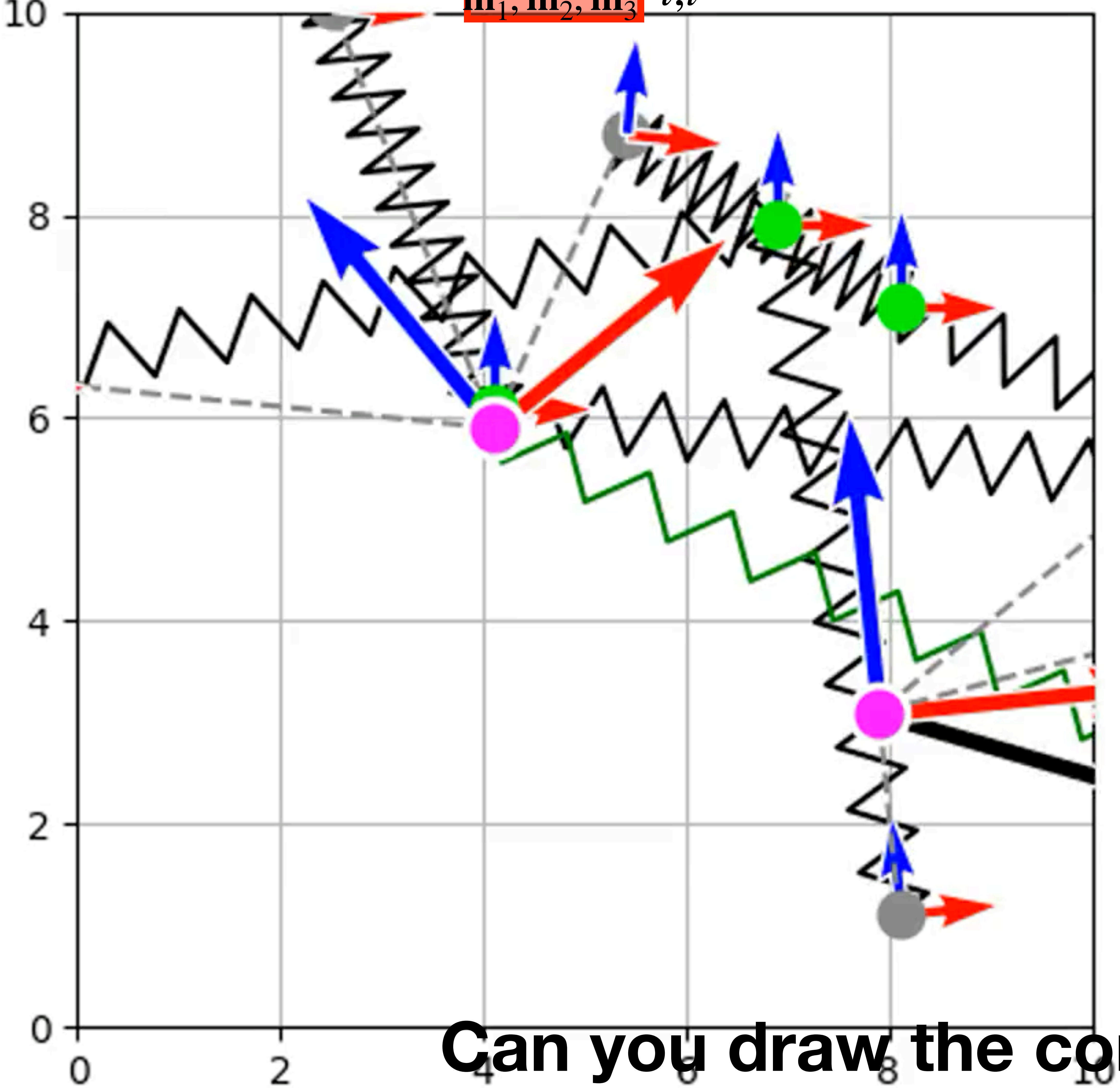
SLAM

3D marker detector (RGBD camera) +



Odometry (IMU)

$$\mathbf{x}_1^\star, \mathbf{x}_2^\star = \arg\min_{\mathbf{x}_1,\mathbf{x}_2,\mathbf{m}_1,\mathbf{m}_2,\mathbf{m}_3} \sum_{i,t} \|\mathrm{w2r}(\mathbf{m}_i, \mathbf{x}_t) - \mathbf{z}_t^{\mathbf{m}_i}\|^2 + \|\mathrm{w2r}(\mathbf{x}_2, \mathbf{x}_1) - \mathbf{z}_{12}^{odom}\|^2$$



$\mathbf{x}_t$ ...robot poses

$\mathbf{m}_i$ ...known marker positions

$\mathbf{z}_t^{\mathbf{m}_i}$ ...marker measurements

$\sum_{i,t} \|\mathrm{w2r}(\mathbf{m}_i, \mathbf{x}_t) - \mathbf{z}_t^{\mathbf{m}_i}\|^2$ ... marker loss

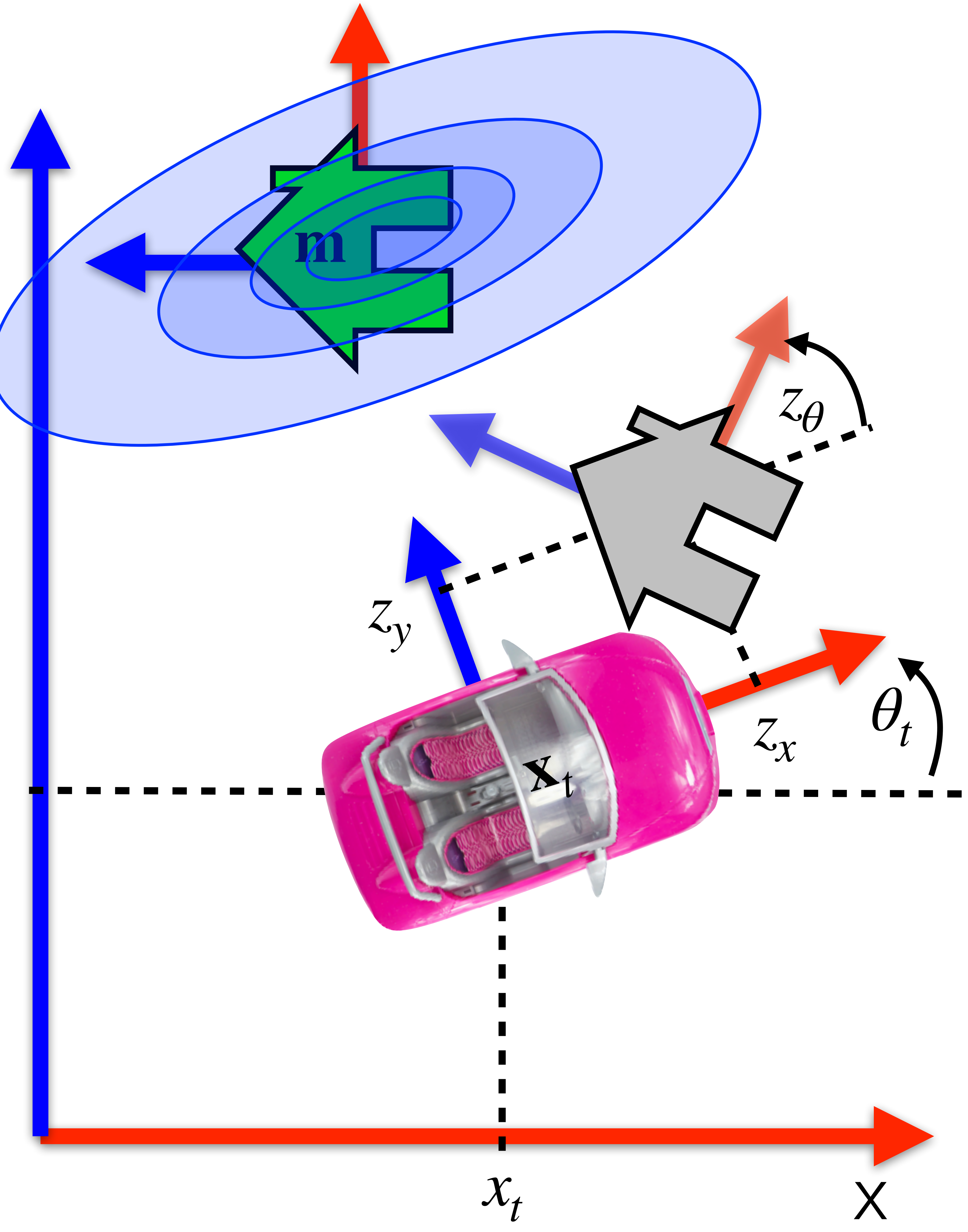
$\|\mathrm{w2r}(\mathbf{x}_2, \mathbf{x}_1) - \mathbf{z}_{12}^{odom}\|^2$ ... odom loss

local coordinate frame

odometry

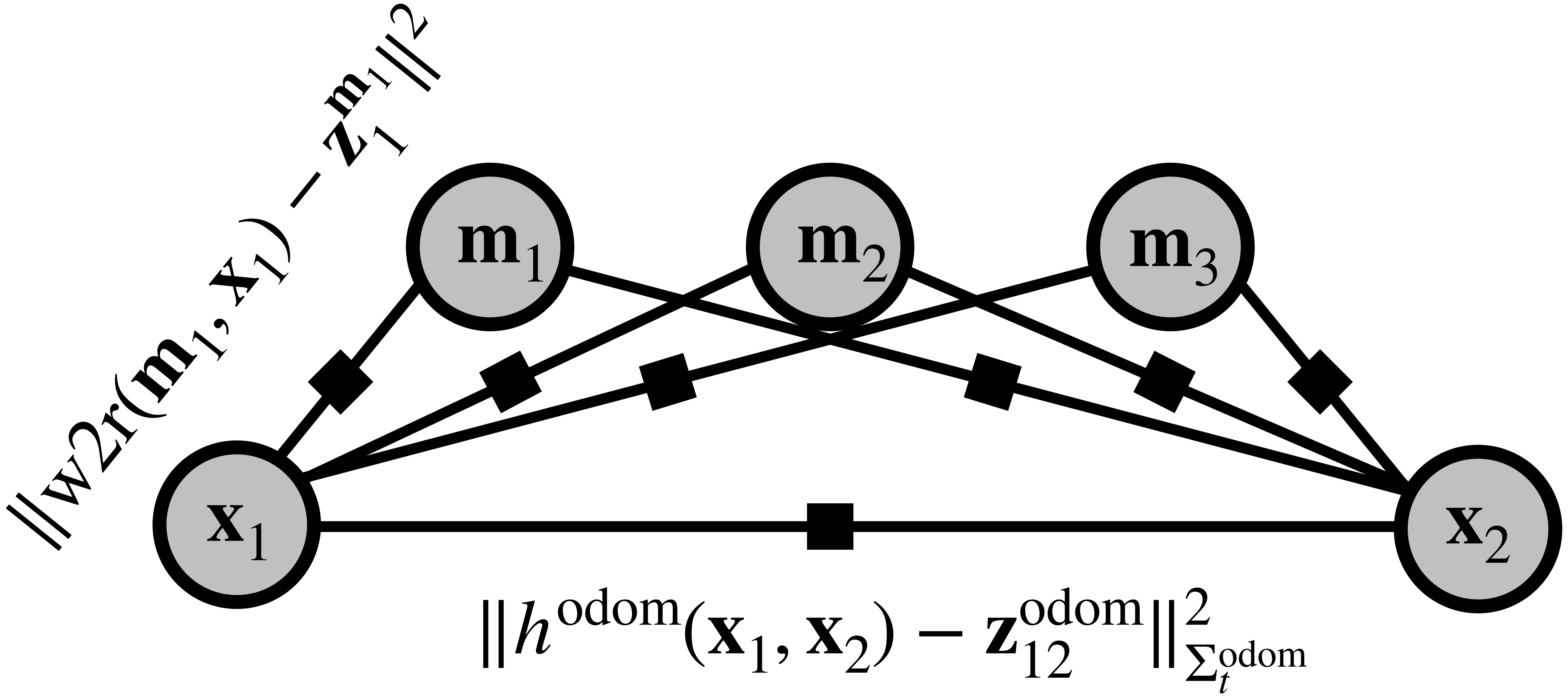
Can you draw the corresponding factorgraph?

Marker detector in EKF SLAM



$$p\Big(\underbrace{\begin{bmatrix} z_t^x \\ z_t^y \\ z_t^\theta \end{bmatrix}}_{\mathbf{z}_t^{\mathbf{m}}} \;\Big|\; \underbrace{\begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix}}_{\mathbf{x}_t}, \underbrace{\begin{bmatrix} m^x \\ m^y \\ m^\theta \end{bmatrix}}_{\mathbf{m}} \Big) = \mathcal{N}\Big(\mathbf{z}_t^{\mathbf{m}};\; \underbrace{\text{w2r}(\mathbf{m}, \mathbf{x_t})}_{h^{\mathbf{m}}(\mathbf{x}_t)} \;,\; Q_t^{\mathbf{m}} \Big)$$

```
loss = ...... # <------- TODO FILL IN
```

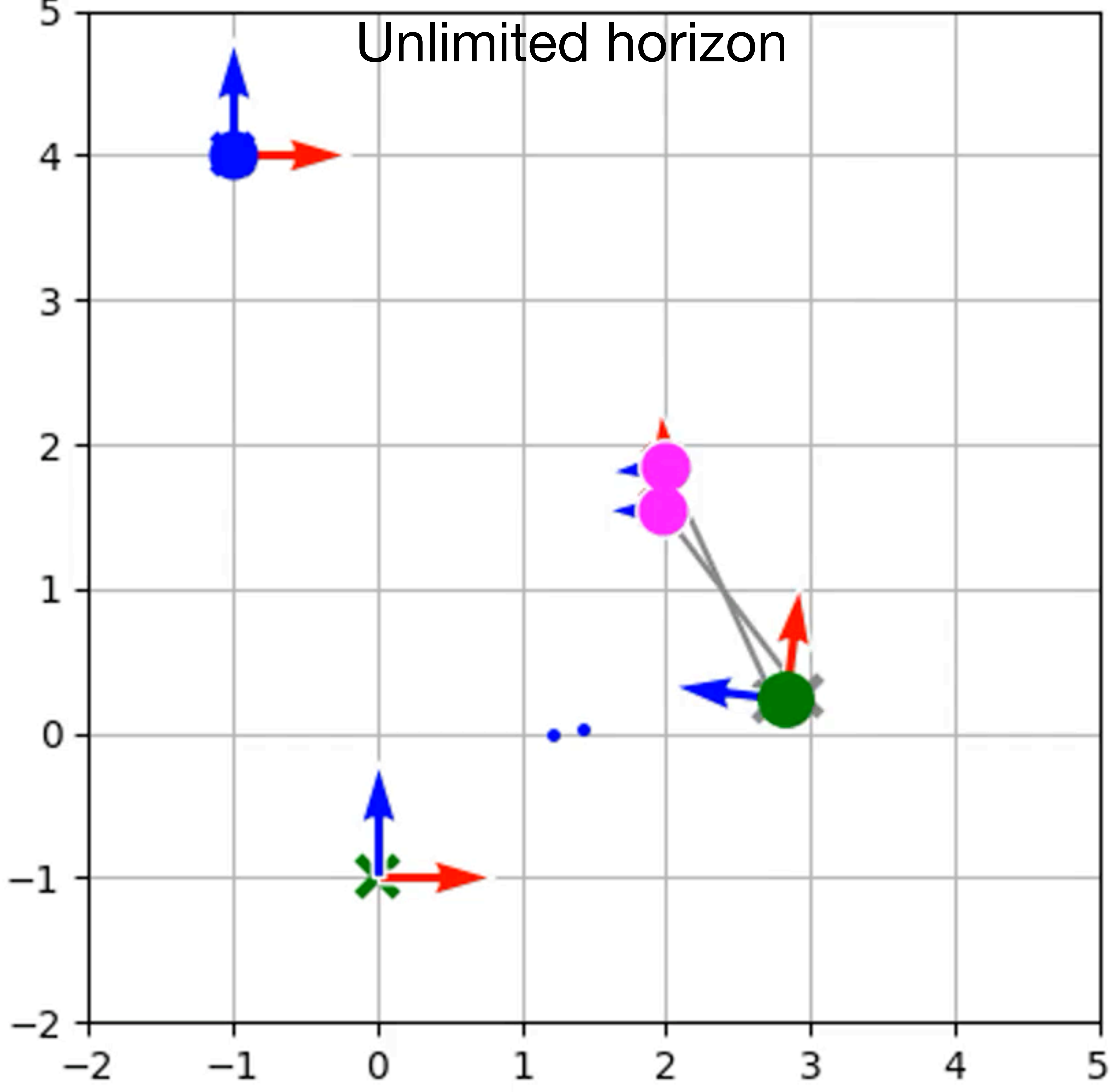


SLAM in SE(2) with successively coming measurements

Combining absolute and relative markers

Combining absolute and relative markers

Combining absolute and relative markers



Combining absolute and relative markers

Limited horizon T=20