McGill
Department of Electrical and Computer Engineering

Indoor Location Estimation for Clinical Applications

Leila Pishdad
Fabrice Labeau

Background

A fundamental requirement for context aware computing is determination of the location. This information is gathered from the location sensors or surveillance cameras. Unfortunately, the sensor measurements have error and need post-processing. Bayesian tracking can be used to estimate the current location from noisy sensor measurements.

Bayesian tracking

In tracking problems, the objective is to estimate the current state, having the current observation. Since the observations are almost always noisy, Bayes filters use probabilistic methods to estimate the current state from the state and measurement equations recursively. In the state equation,

$$\mathbf{x}_t = \mathbf{f}_t(\mathbf{x}_{t-1}, \mathbf{u}_t), \qquad (1)$$

the system transition function $\mathbf{f}_t$ relates the current state $\mathbf{x}_t$ to the previous state $\mathbf{x}_{t-1}$ and the current process noise $\mathbf{u}_t$. The current measurement $\mathbf{z}_t$ is related to the current state $\mathbf{x}_t$ and the current noise vector $\mathbf{v}_t$ by the measurement function $\mathbf{g}_t$ as:

$$\mathbf{z}_t = \mathbf{g}_t(\mathbf{x}_t, \mathbf{v}_t). \qquad (2)$$

The system transition function $\mathbf{f}_t$ and the measurement function $\mathbf{g}_t$ can be nonlinear. Additionally, the noise vectors can be taken from any known probability distributions [1]. Using Bayesian tracking, the posterior density $p(\mathbf{z}_t|\mathbf{x}_t)$ is evaluated. In location estimation problems, the state vector $\mathbf{x}_t$ contains the location information, which can be simply the coordinates or could also include the velocities, etc. The motion model and the sensor model provide details on Eq. (1) and Eq. (2), respectively [2,3].

Motion model

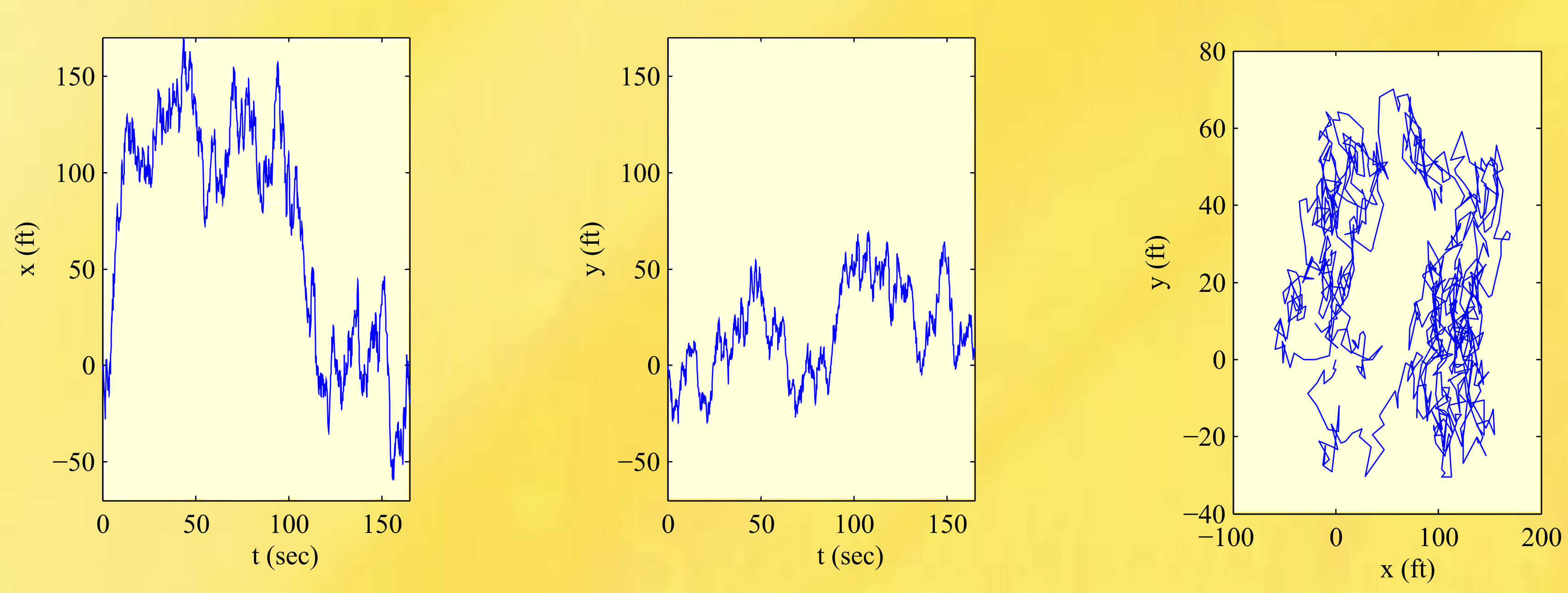
The motion model used for our experiment is defined as:

$$p\left(\mathbf{X}_t = \mathbf{x}_t | \mathbf{X}_{t-1}^i = \mathbf{x}_{t-1}^i\right) \sim \mathrm{N}\left(\mu_{t|t-1}, \sigma_{t|t-1}^2\right).$$

which is evaluated assuming that at each iteration white noise $n_t \sim \mathrm{N}(\mu_n, \sigma_n^2)$ is added to the velocity:

$$\mathbf{v}_t = \mathbf{v}_{t-1} + \mathbf{n}_t.$$

The following figures show a sample trajectory generated by this model.



a) x direction vs. time

b) y direction vs. time

c) x vs. y direction

Figure 1 : a sample trajectory

Sensor model

Assuming:

$$\mathbf{z}_t = \mathbf{x}_t + \mathbf{e}_t.$$

where $\mathbf{e} = < \mathrm{e_x}, \mathrm{e_y} >$, as the measurement equation, we have $\mathbf{e}_t = \mathbf{z}_t - \mathbf{x}_t$ and $p(\mathbf{z}_t|\mathbf{x}_t)$ can be computed as $p(\mathbf{z}_t|\mathbf{x}_t) = p(\mathbf{e}_t = \mathbf{z}_t - \mathbf{x}_t)$, by using the sensor measurements for stationary tags. Using Gaussian Mixture Model with c clusters in each direction, the sensor error distribution can be written as:

$$p(e) = \sum_{j=1}^{c} f\left(x; \mu_j, \sigma_j^2\right),$$

in each direction, where e could represent either e_x or e_y.

Particle filters

Particle filters use samples to approximate the posterior distribution recursively. The samples are taken from a proposal distribution $\pi(x)$ and are associated with weights to approximate the density $p(x)$. Using this approximation $p(x)$ can be approximated as:

$$p(x) \approx \sum_{i=1}^{N_s} w^i \, \delta\left(x - x^i\right),$$

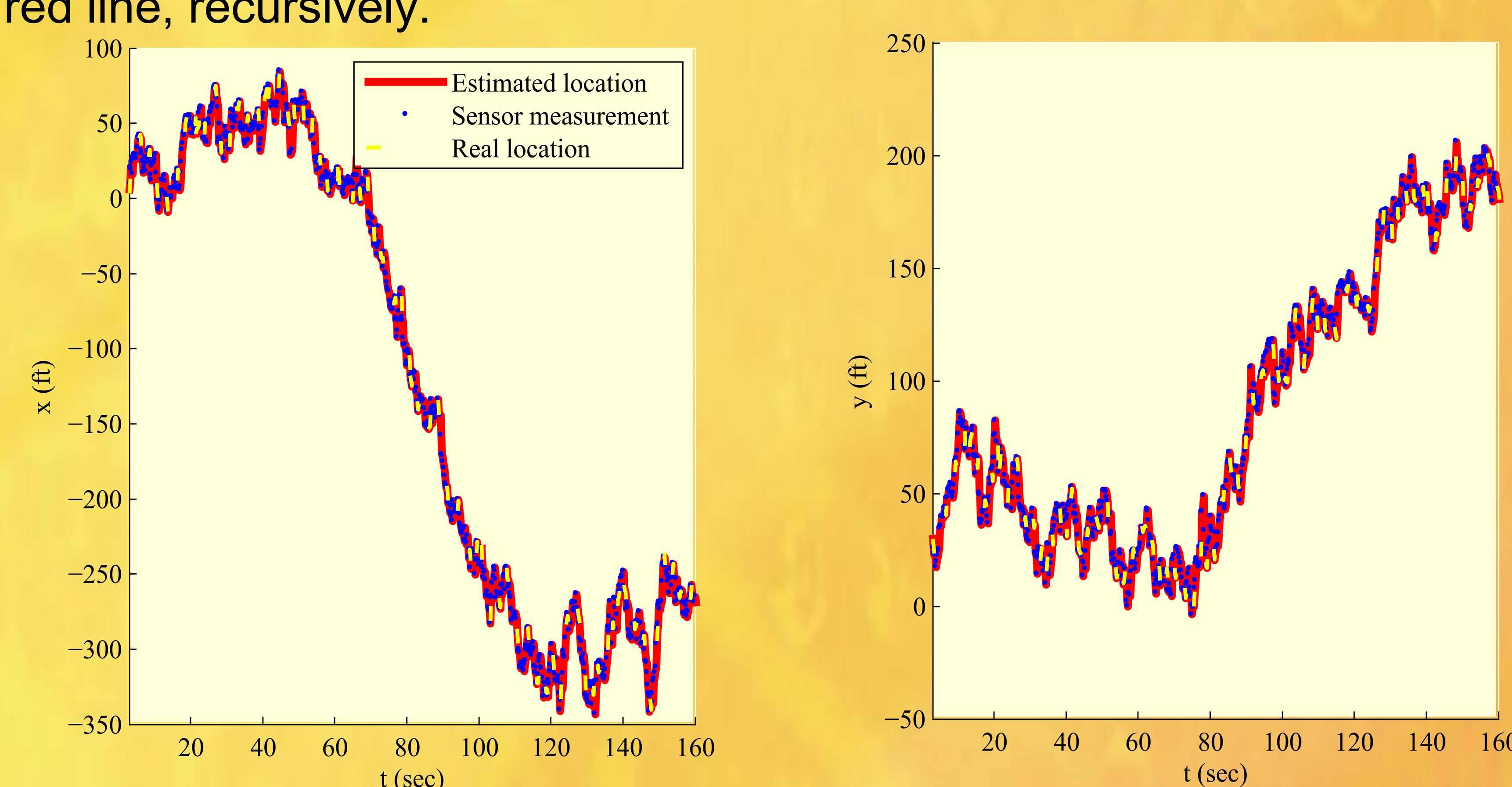
where N_s is the number of samples and

$$w^i \propto \frac{p(x)}{\pi(x)}.$$

The proposal distribution $\pi(x)$ is called the importance function. As the importance density approximates the posterior, a better choice for it could improve the performance [4]. There are many choices for the importance function among which the prior importance function, $\pi(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{z}_{0:t}) = p(\mathbf{x}_t|\mathbf{x}_{t-1})$, and the optimal importance function, $p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t)$ are the most common.

Simulation results

Using the provided motion model and sensor model, the optimal importance function can be evaluated. In the figure below, the real location, sensor measurements, and estimated location (using particle filtering with optimal importance function), are represented by yellow dashed line, blue dots, and red line, recursively.



a) x vs. time

b) y vs. time

Figure 2 : Location estimation with MSSI sensors for synthetic data

The table below shows a comparison between the two importance functions. The performance metrics used are the root of Mean Squared Error, and maximum error in each direction.

x		y		Importance Function
RMSE	Max error	RMSE	Max error	
2.6328	18.4104	2.3916	19.792	Prior
0.5414	1.4110	0.3130	1.1655	Optimal

References

[1] P. M. Djuric, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," IEEE Signal Processing Magazine, vol. 20, pp. 19–38, September 2003.

[2] D. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello, "Bayesian filtering for location estimation," IEEE Pervasive Computing, vol. 2, pp. 24–33, July-September 2003.

[3] J. Hightower and G. Borriello, "Particle filters for location estimation in ubiquitous computing: A case study," UbiComp 2004: Ubiquitous Computing, pp. 88–106, 2004.

[4] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," IEEE Transactions on Signal Processing, vol. 50, pp. 174–188, Feb 2002.