Tracking Systems
Particle Filter

Mayukh Sattiraju
msattir@clemson.edu

ECE 854
Lab 6 Report

November 21, 2017

1 Introduction

This lab deals with application of a Particle filter. The velocity update for the filter is non-differentiable. Thus, to track this problem, we would need to employ a Particle Filter.

2 Methods

The implementations of the Particle Filter and the re-sampling method is given below.

2.0.1 1D Tracking

The state space equations that describe the position and velocity of a 1D particle is given as

$$f(x_t, a_t) = \begin{bmatrix} x_{t+1} = x_t + \dot{x}_t T \\ \dot{x}_{t+1} = \begin{cases} 2 & x_t < -20 \\ \dot{x}_t + |a_t| & -20 \leq x < 0 \\ \dot{x}_t - |a_t| & 0 \leq x \leq 20 \\ -2 & -20 \leq x < 0 \end{cases} \end{bmatrix}$$

The system measures the magnitude of a magnetic pulse at different positions. This input measurement is taken from the data provided. The ideal mesaurement of the particle is defined as (a function of the static positions of the two magnets and the dynamic position of the object):

$$y_t^{(m)} = \frac{1}{\sqrt{2\pi}\sigma_m}\left(\exp\left(\frac{-\left(x_t{}^{(m)} - x_{m1}\right)^2}{2\sigma_m^2}\right) + \exp\left(\frac{-\left(x_t{}^{(m)} - x_{m2}\right)^2}{2\sigma_m^2}\right)\right)$$

The particles' weight update equation uses the particle's distance, models it to a measurement and checks the offset of this measurement from the input measurement. Based on the magnitude of this offset, the particle's weight is updated. This is done for each particle.

The probability function used for weight updation is given as:

$$p\left(y_t | \, x_t^{(m)}\right) = \frac{1}{\sqrt{2\pi}\sigma_n}\exp\left(\frac{-\left(y_t{}^{(m)} - y_t\right)^2}{2\sigma_n^2}\right)$$

2.1 Resampling

Re-sampling is done to evolve the intelligence of the set. This removes particles that are no longer relevant to the tracking and replaces them in positions of better particles.

Resampling can be done in many ways. Here's the provided sudo code for Resampling

Listing 1: Re-Sampling Pseudo Code

```
Assume particle states in P[1...M], weights in W[1...M].
Q=cumsum(W); calculate the running totals
t=rand(M+1); t is an array of M+1 uniform random numbers 0 to 1
T=sort(t); sort them smallest to largest
T[M+1]=1.0; boundary condition for cumulative hist
i=j=1; arrays start at 1
while (i<=M)
  if (T[i] < Q[j])
    Index[i]=j;
    i=i+1;
  else
    j=j+1;
  end if
end while

loop (i=1; i<=M; i=i+1)
  NewP[i]=P[index[i]];
  NewW[i]=1/M;
end loop
```

3 Results

The parameters used for this implementation are:

- ${\sigma_a}^2 = 0.01$
- $\sigma_n = 0.004$
- $\sigma_m = 4$
- M=100
- Re-sampling is done when ESS falls below 20%

Here's a result of tracking the position

Figure 2 shows a zoom of the plot. The measurement data is the value of the measured magnetic strength, while the estimated and actual states are positions. As the scales' of the quantities are different the measurement values seem to be at 0.

Figure 3 is another tracking, where the actual and estimated positions are out-of-phase.

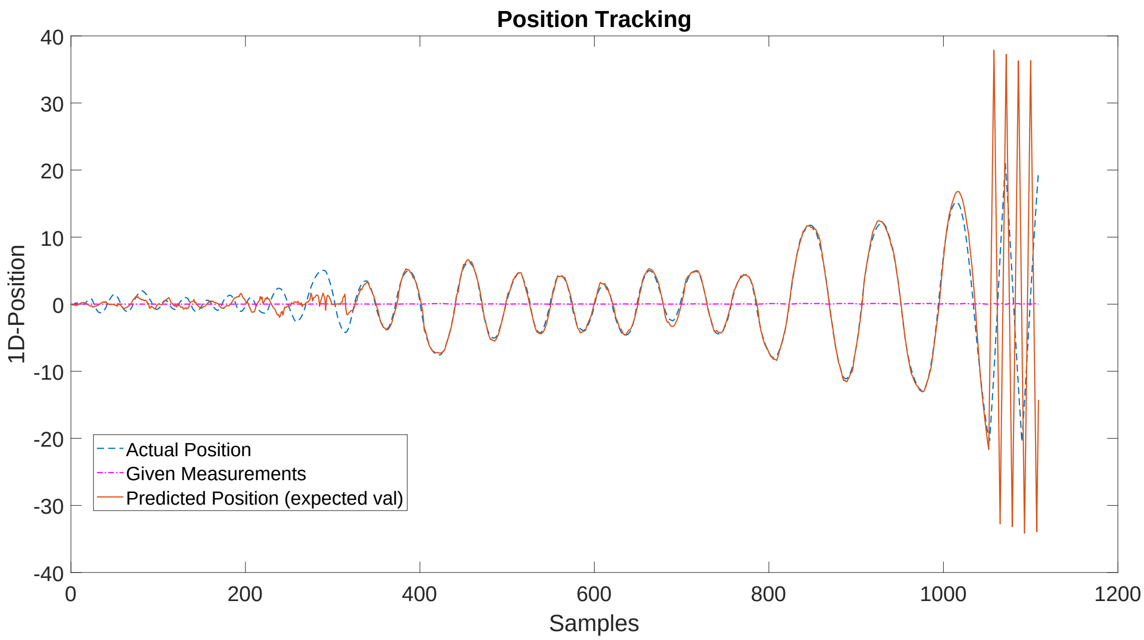


Figure 1: Particle Filter Position Tracking

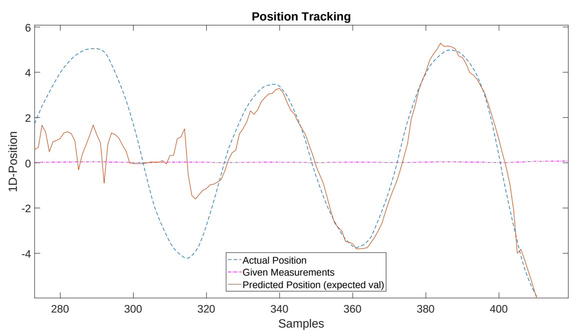


Figure 2: Zoom of Position Tracking

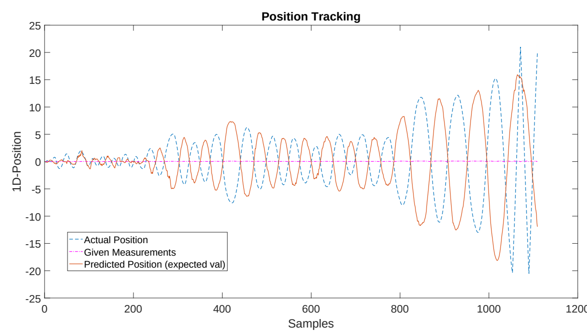
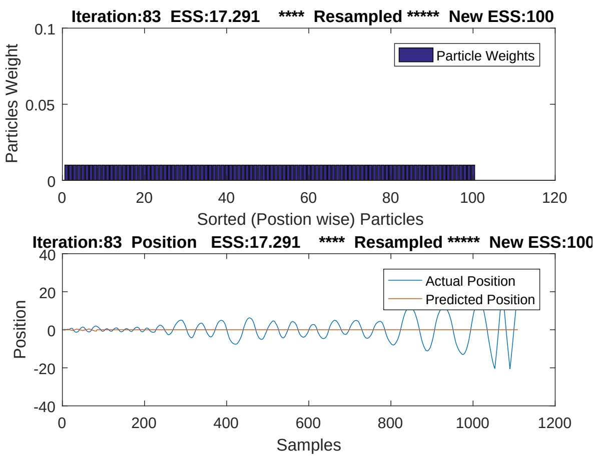
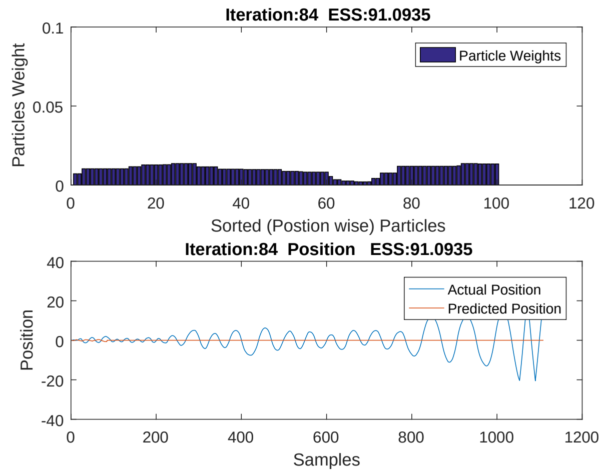
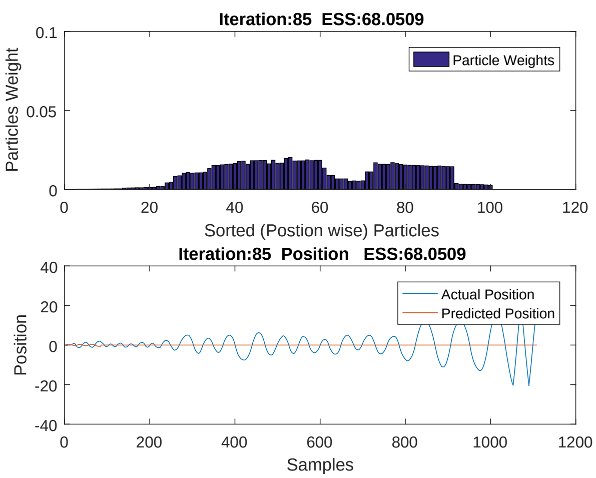


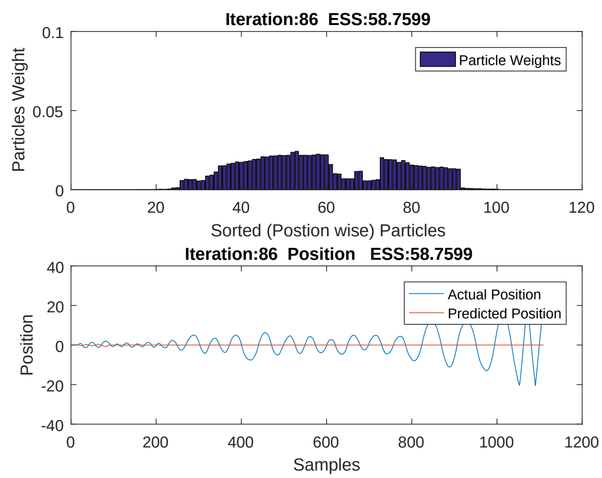
Figure 3: Out-of-Phase Tracking

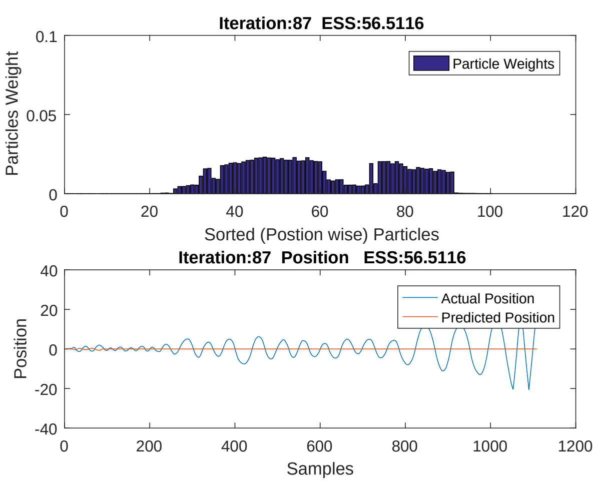
The distribution of weights of the particle through iterations is shown in Figure 4

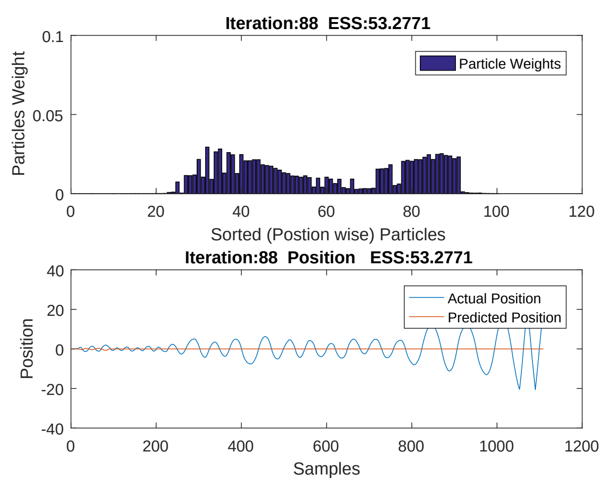

Iteration:83 ESS:17.291 **** Resampled ***** New ESS:100
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:83 Position ESS:17.291 **** Resampled ***** New ESS:100
Actual Position
Predicted Position
Position
Samples

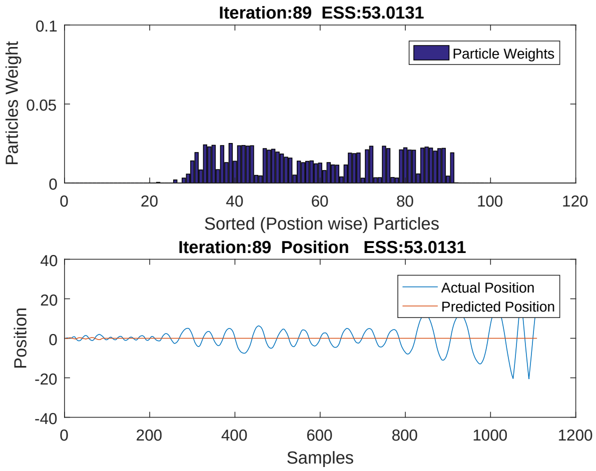

Iteration:84 ESS:91.0935
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:84 Position ESS:91.0935
Actual Position
Predicted Position
Position
Samples

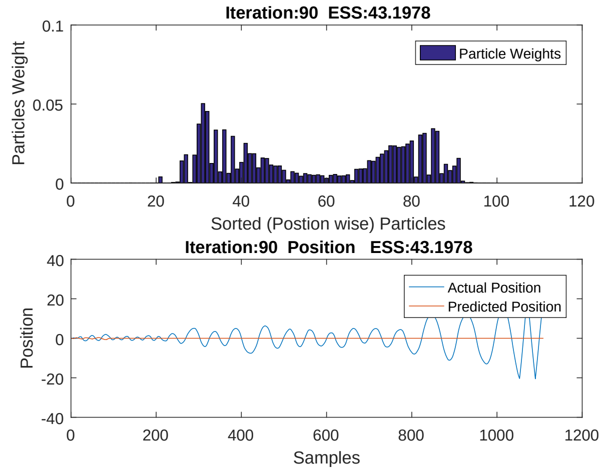

Iteration:85 ESS:68.0509
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:85 Position ESS:68.0509
Actual Position
Predicted Position
Position
Samples

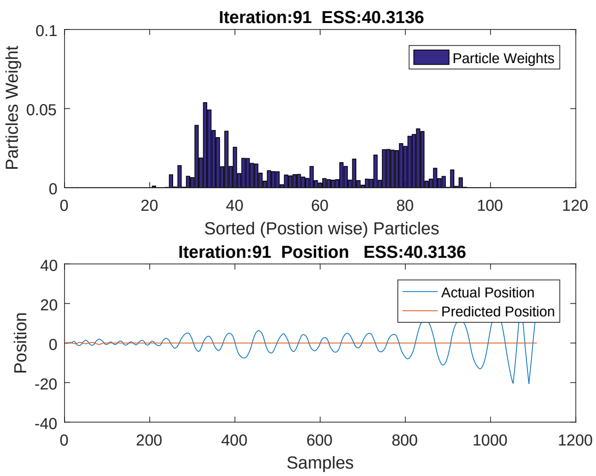

Iteration:86 ESS:58.7599
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:86 Position ESS:58.7599
Actual Position
Predicted Position
Position
Samples

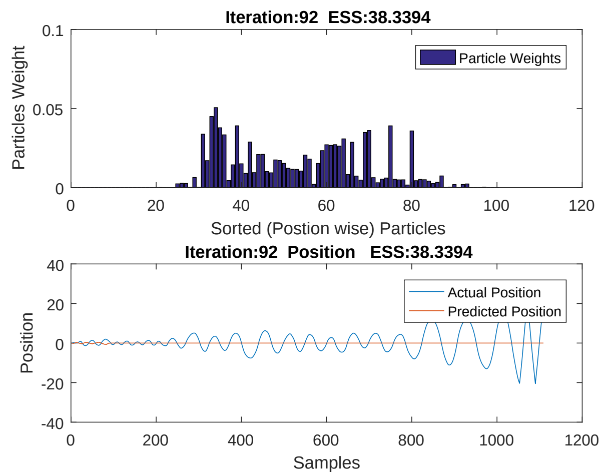

Iteration:87 ESS:56.5116
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:87 Position ESS:56.5116
Actual Position
Predicted Position
Position
Samples

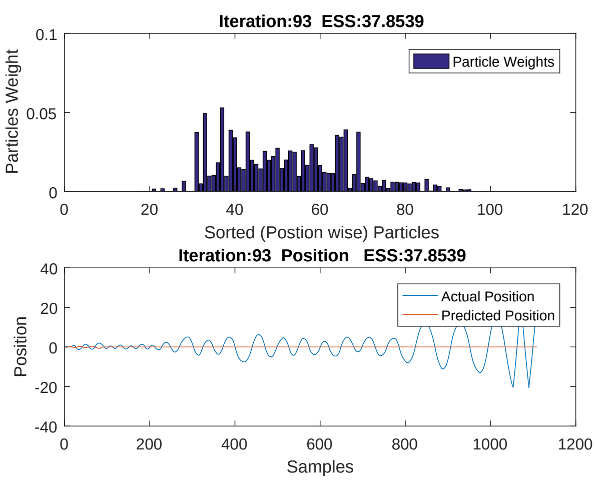

Iteration:88 ESS:53.2771
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:88 Position ESS:53.2771
Actual Position
Predicted Position
Position
Samples

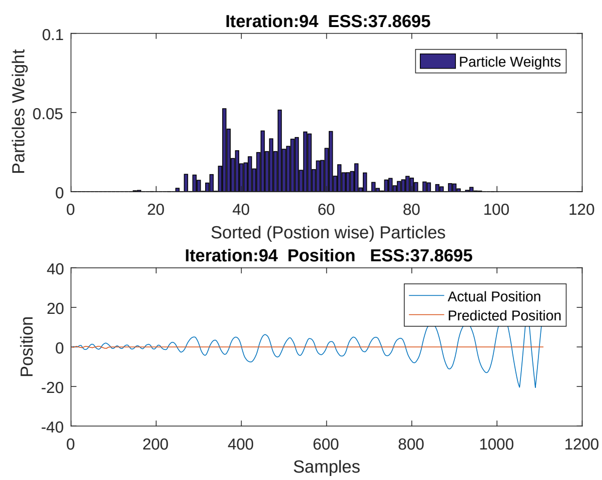

Iteration:89 ESS:53.0131
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:89 Position ESS:53.0131
Actual Position
Predicted Position
Position
Samples


Iteration:90 ESS:43.1978
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:90 Position ESS:43.1978
Actual Position
Predicted Position
Position
Samples


Iteration:91 ESS:40.3136
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:91 Position ESS:40.3136
Actual Position
Predicted Position
Position
Samples


Iteration:92 ESS:38.3394
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:92 Position ESS:38.3394
Actual Position
Predicted Position
Position
Samples


Iteration:93 ESS:37.8539
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:93 Position ESS:37.8539
Actual Position
Predicted Position
Position
Samples


Iteration:94 ESS:37.8695
Particle Weights
Particles Weight
Sorted (Postion wise) Particles
Iteration:94 Position ESS:37.8695
Actual Position
Predicted Position
Position
Samples

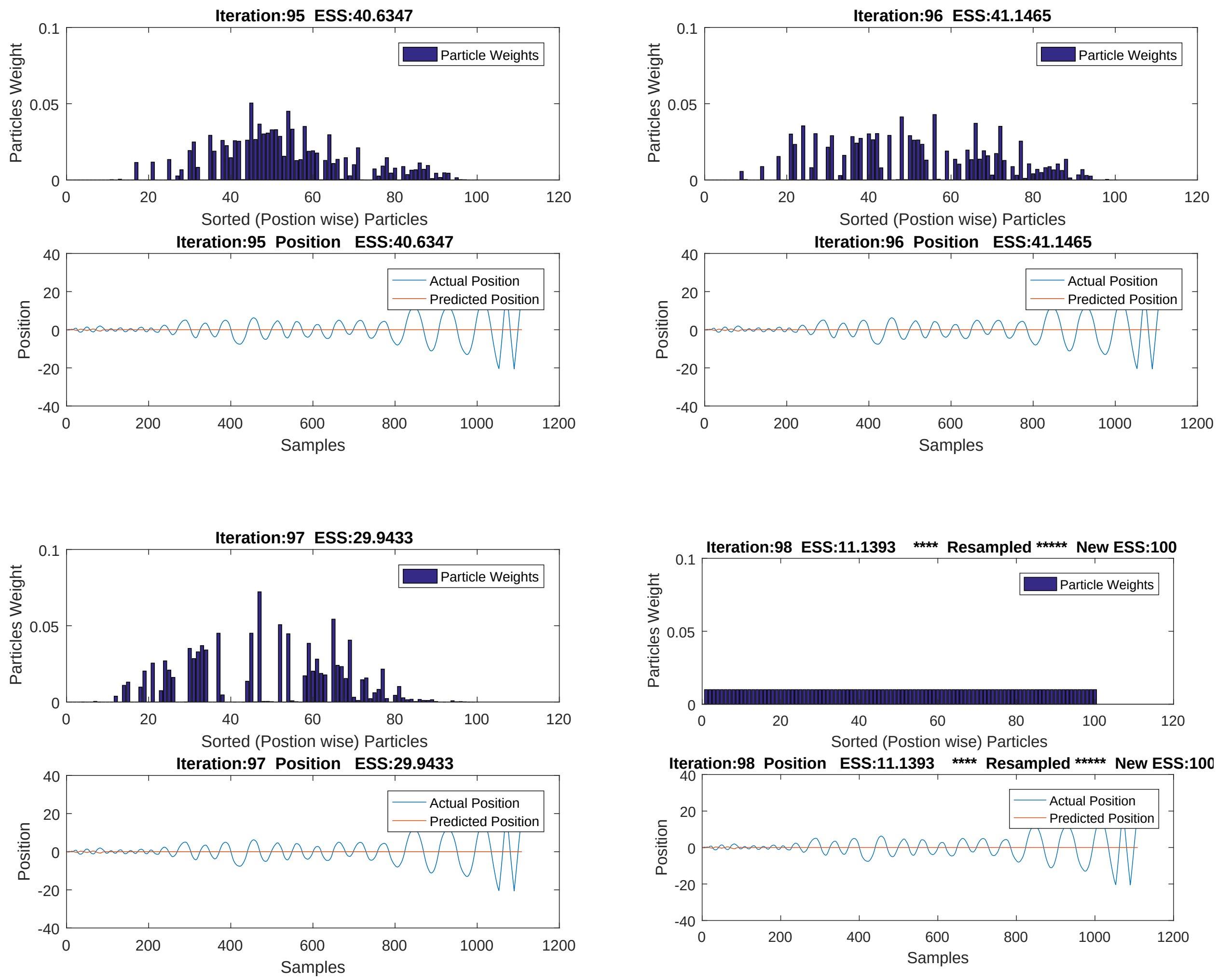


Figure 4: Particles' Weights Histogeams