Report Particle Filter

Francesco Mecatti
299822@studenti.unimore.it

This report lists four different solutions, from the most naive to the most complex one. In this assignment, three main components of the Particle Filter (PF) have been identified and tweaked:

- initialization function
- observations association algorithm
- resampling strategy

1 Random Init |NN Association |Multinomial Resampling

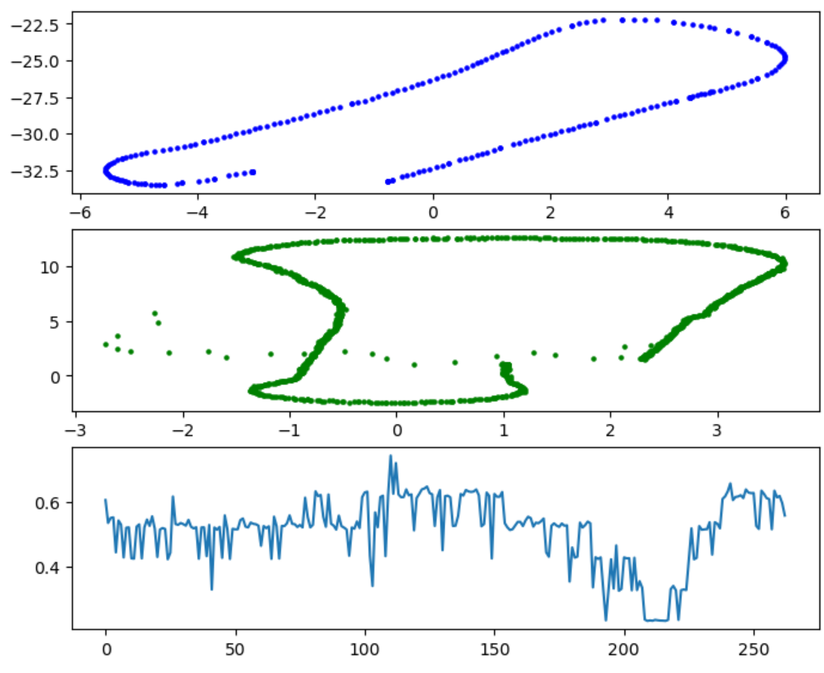

Figure 1: Trajectories w/ Random Initialization, Nearest Neighbor Association, and Multinomial Resampling

With this solution, a random initialization strategy, in which initial particles are uniformly distributed within the map, proved to underperform wrt initialization with initial guess. It can be observed that a wrong pose picked as a starting point introduces an error that propagates on the subsequent localizations. This results in a stretched and rotated (due to the wrong initial heading assumption) path, quite different from the ground truth, in terms of proportions and orientation.

2 Initial Guess |NN Association |Multinomial Resampling

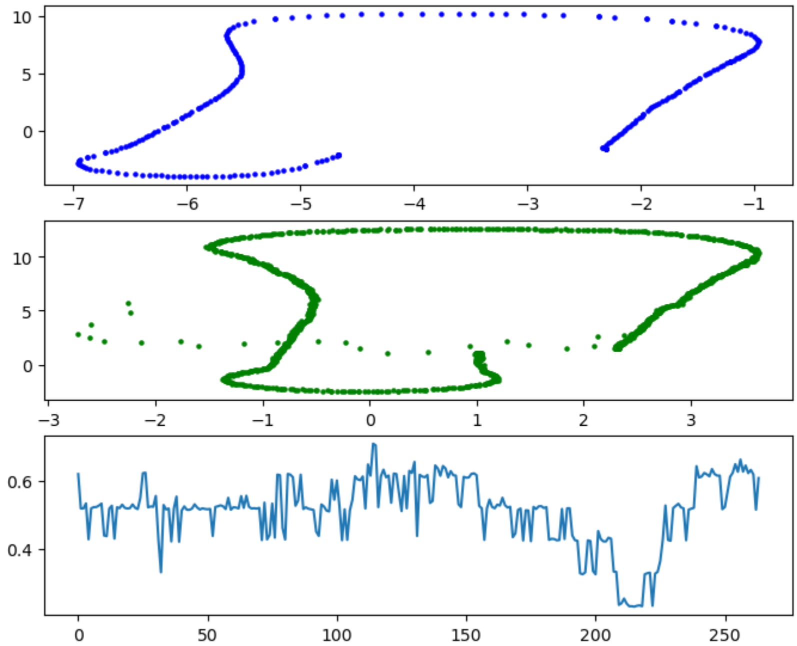

Figure 2: Trajectories w/ Initial Guess Initialization, Nearest Neighbor Association, and Multinomial Resampling

Instead, in this experiment, the Particle Filter is initialized with a known position, the *initial guess*, meaning that all particles are generated around its location.

Contrary to the previous case, in which the location of the particles was drawn from a uniform distribution, ensuring diversity by design, this time the initialization noise is the only means that guarantees diversity between the initial particles. The initialization noise is quantified by the `sigma_init` variable.

Overall, the path looks more similar to the reference, even though the scale of the X axis is still messed up (stretched in the bottom and shrunk in the top).

While experimenting with parameters tuning, it emerged that this configuration is more robust to noisy predictions than the first one. It is probably due to the fact that even in case of model drifting (and consequent discrepancy with real forklift motion), there are enough particles around the real location to recover a good pose estimation with the following update. In this experiment, the prediction variance has been increased from 0.01 to 0.3 for each of the three dimensions: x, y, and theta.

3 Initial Guess |Hungarian Association |Multinomial Resampling

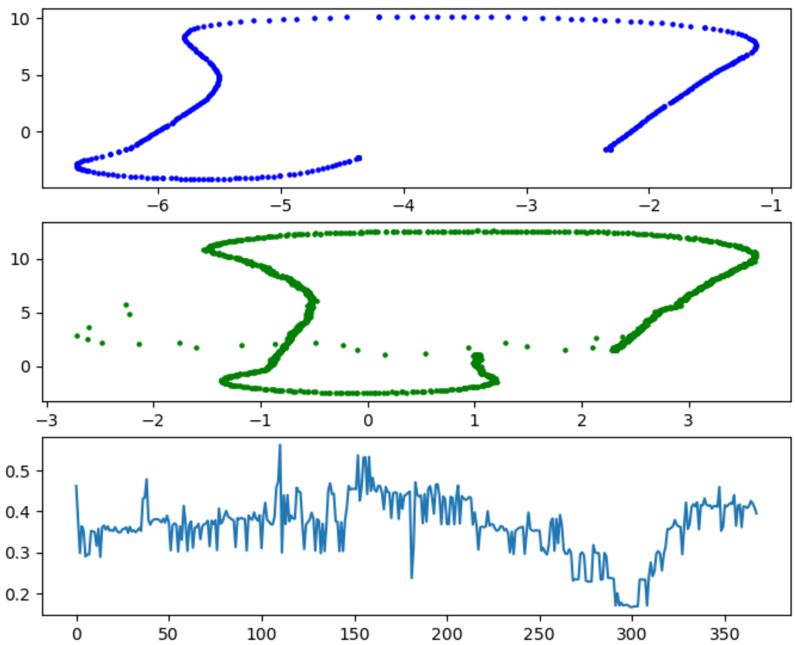

Figure 3: Trajectories w/ Initial Guess Initialization, Hungarian Algorithm Association, and Multinomial Resampling

Here, a different association algorithm has been tested: instead of a nearest-neighbor (NN) search, the Hungarian (Munkers) algorithm is used. The reason is that NN leads to potentially suboptimal or conflicting associations. The outcome depends on how the algorithm is implemented. If chosen elements are discarded after an association is established, the solution optimality cannot be ensured due to the greedy formulation of the algorithm: depending on the processing order, an element could be starved of its best association candidate because another element has already been associated to it. On the other hand, if associated elements are not discarded, conflicts may arise: two or more elements of the first set associated to the same element of the second set. Both suboptimality and conflicts may fake out weighting scores.

The Hungarian algorithm solves the assignment problem optimally. By filling the cost matrix with the distance between each pair of elements, the algorithm will minimize the total cost of associations, finding a globally optimal solution.

The path obtained from this experiment has a slightly better scale on the X axis than the previous ones. In this experiment, particles number has been increased from 1k to 5k, leading to a longer execution time, as shown in 3 (350 vs 250 time steps).

4 Initial Guess |Hungarian Association |Stratified Resampling

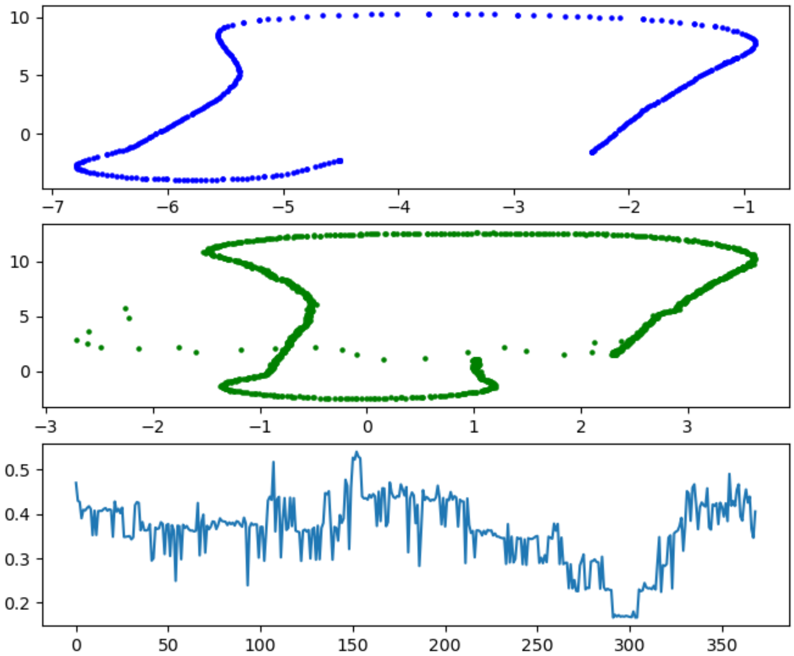

Figure 4: Trajectories w/ Initial Guess Initialization, Hungarian Algorithm Association, and Stratified Resampling

This solution introduces a stratified resampling strategy to get around multinomial sampling "unfairness". Stratified resampling promotes diversity, which should be beneficial in order to find an ever better approximation of the real forklift's position.

Stratified resampling has been implemented by normalizing the weights and computing a cumulative probability (weight) function. Then the probability interval (0, 1) is divided into slices of equal length. Eg. 0-0.1, 0.1-0.2, ..., 0.9-1.0

A draw from each slice is ensured: for each of them a uniform distribution draw in the slice interval is performed. The respective particle is thus extracted from the draw.

The results are generate using the same parameters as in the previous experiment. The overall path is very similar to the previous one, except for the X-axis scale, which looks like it has worsened.