Trajectory and parameter estimation using Sequential Monte Carlo methods and Expectation Maximization

Xavier Svensson Depraetere
Markus Meder

Abstract

The focus of this research was twofold.
- To explore performance differences between a particle filter and a particle smoothing technique in a specific tracking problem. The particle filter technique used is Sequential Importance Sampling with Resampling (SISR). The particle smoothing technique used is Fixed Lag Smoothing (FLS).
- To explore performance differences in parameter estimation, using Expectation-Maximization (EM) in conjunction with SISR and FLS.

The findings were that the FLS algorithm outperformed the SISR algorithm in both target trajectory and parameter estimations. However, the difference in parameter estimations and the convergence rate was found to be very small. In conclusion we recommend using FLS for target trajectory estimates for this and similar problems, but the choice might not be as clear cut when comparing parameter estimation performances.

Problem statement

Our project will explore tracking and parameter estimation in a non-linear cellular network through the use of sequential Monte-Carlo (SMC) smoothing. The aim is twofold. Firstly we want to compute a smoothed trajectory estimation of a target given signal strengths collected at several base stations, see figure 1. This is achieved by using the Fixed Lag Smoothing technique (FLS). The results are then compared to a non-lagged based technique called Sequential Importance Sampling with Resampling (SISR). Secondly we aim to improve the underlying model using Estimation Maximization (EM). The parameters which are to be determined is the signal falloff η and the signal variance ξ². We employ a hidden Markov Model (HMM) to model this.

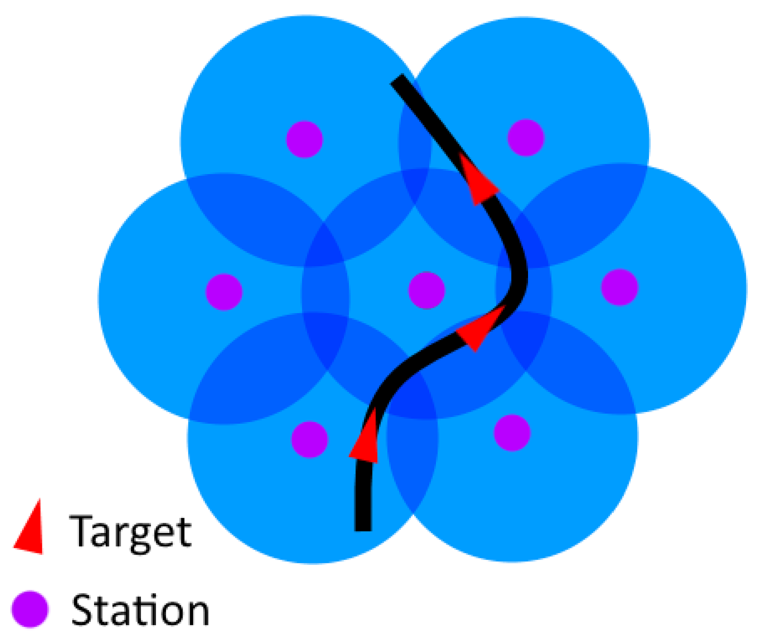

Figure 1: An illustration of the tracking problem.

Method

The trajectory is modeled using a Hidden Markov Model (HMM). The underlying Markov model is describing the movement of the target whilst the output distribution describes the observed signal strength of the target. The output distribution is given by

$$Y_n^l = v - 10\eta \log_{10}(||(X_n^1, X_n^2) - \pi_l||) + V_n^l$$

where ν is the base signal strength, η is the signal strength fall-off, (X^1, X^2) is the position of the target at time n and π_l is the position of the l'th base station. Furthermore, V^l_n is a noise term with variance ξ².

SISR is a method of approximating the conditional probability of the states of the underlying Markov process. Firstly, particles are simulated from an HMM. Weights are then assigned to these particles proportionally to the likelihood of the simulated state producing the observed output. A state estimation can then be determined by summing the product of the normed weights with their respective particles. The particles are then resampled with probability proportional to their associated weights.

FLS is a smoothing algorithm. Meaning that it aims to make estimates by making use of past, present and future data points. In this application it is similar to SISR however the resampling is preformed with regards to the future weights of the particles rather than the current.

The EM algorithm is an iterative method for finding maximum likelihood estimations of a parameter. This is done by first creating a function of the log-likelihood using current estimates of the parameters. Secondly, the function is then maximized with regards to the parameters in interest, giving a new estimate of the parameters.

In order to test and compare the methods we simulate one data set from the HMM with the fall-off parameter η = 3 and noise variance ξ² = 2.25. The observation sequence of this data set will then be used as input for the two methods. The difference in tracking preformance will be compared for 500 time lengths via plots and error analysis. The maximum likelihood parameter estimates will be compared with regards to three different timelengths and for different lags in the FLS method. We will also investigate the convergence rate for both methods by plotting several runs of the algorithms.

Results trajectory estimation

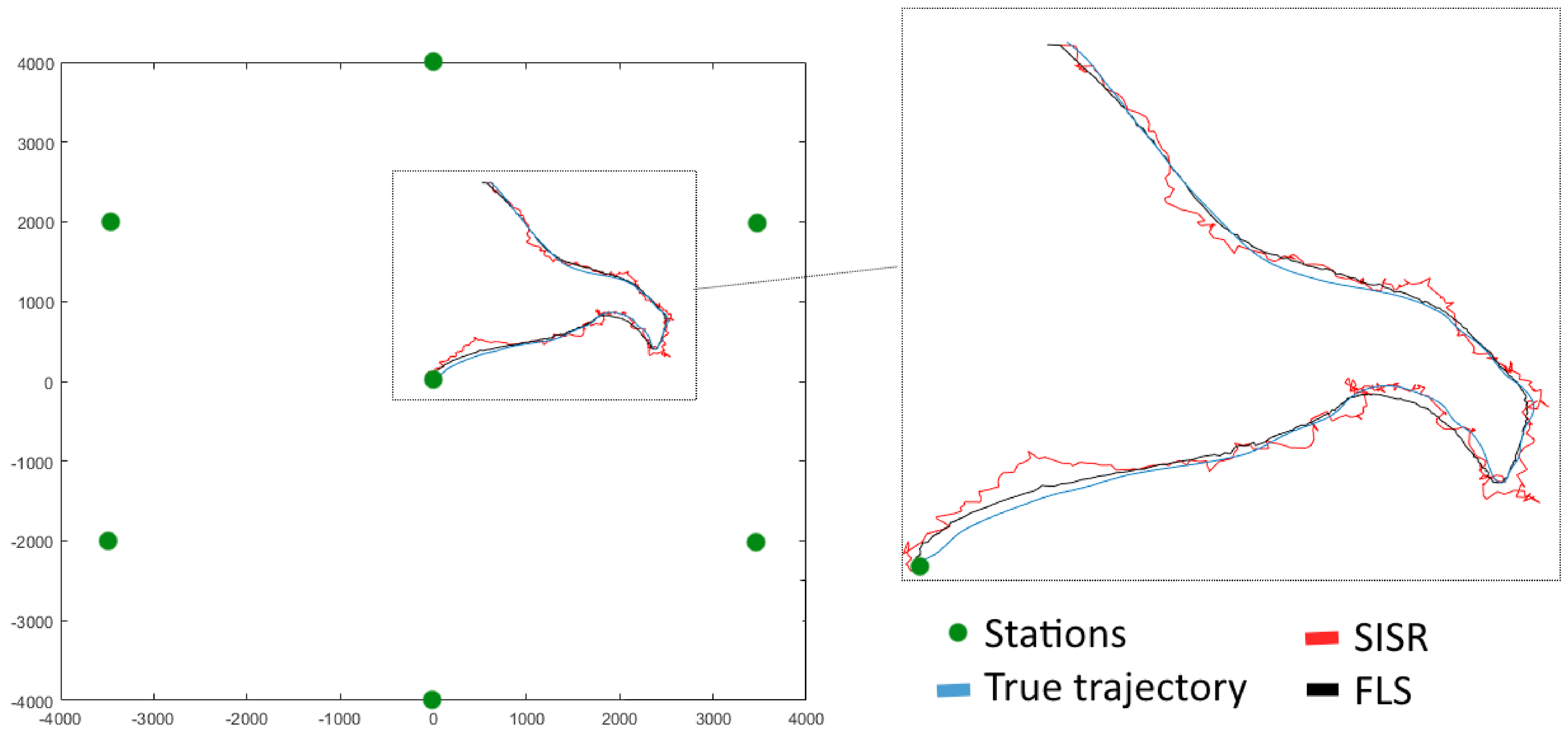

Figure 2: True trajectory of an object with base station positions and trajectory estimations.

n	Lag	0	2	4	8	16	32
100	MSE [10^3]	2.6889	2.9076	2.1229	1.9705	1.5597	1.7244
	Var [10^5]	1.1732	0.0345	0.0250	0.0247	0.0158	0.0211
200	MSE [10^3]	3.7606	4.1377	2.6875	1.6736	1.4928	1.4561
	Var [10^5]	8.7589	0.0383	0.0257	0.0159	0.0141	0.0154
500	MSE [10^4]	3.8606	3.3488	3.0624	2.1807	1.2417	0.5894
	Var [10^7]	4.3875	0.0705	0.0679	0.0492	0.0261	0.0077

Table 1: Mean square error (MSE) and variance of squared error of simulations of different time lengths. From top to bottom these lengths are 100, 200 and 500.

Results parameter estimation

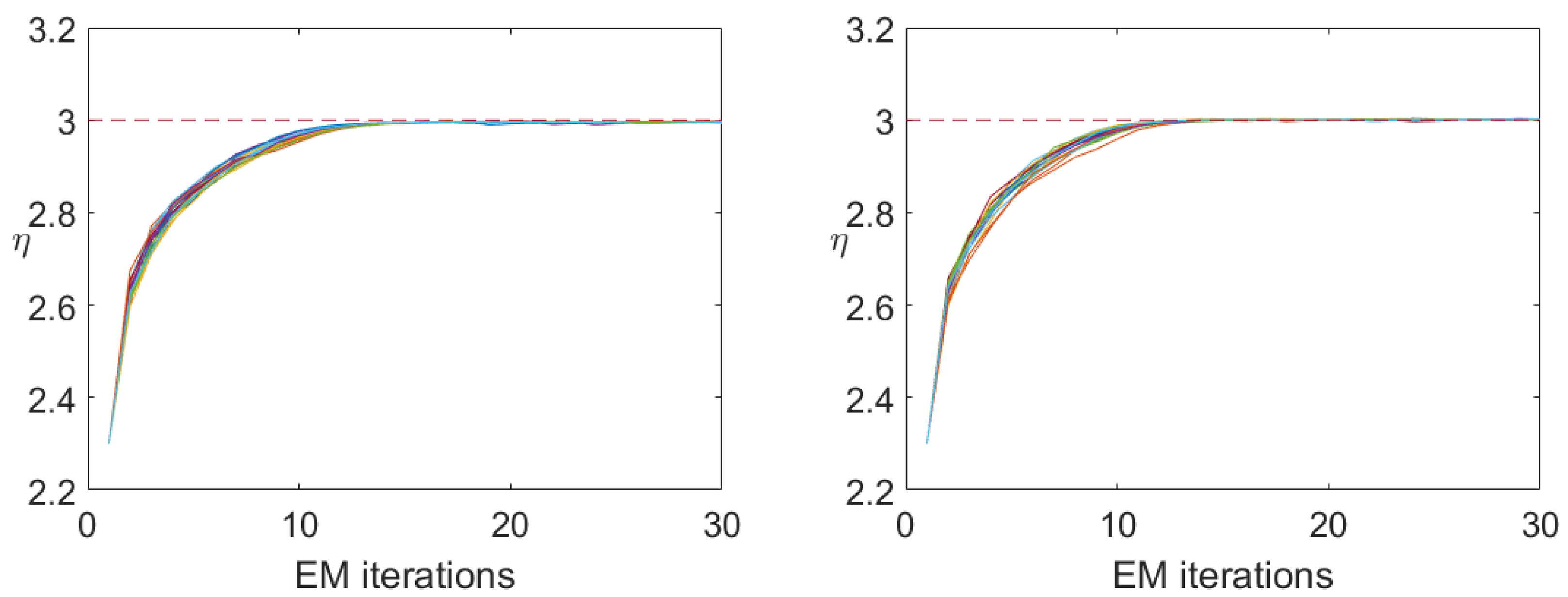

Figure 3: Convergence of estimations of η for 20 runs of time length 500. The left figure show the SISR estimates and the right figure show the FLS estimates for lag 32. The true value of η is marked by the dotted line.

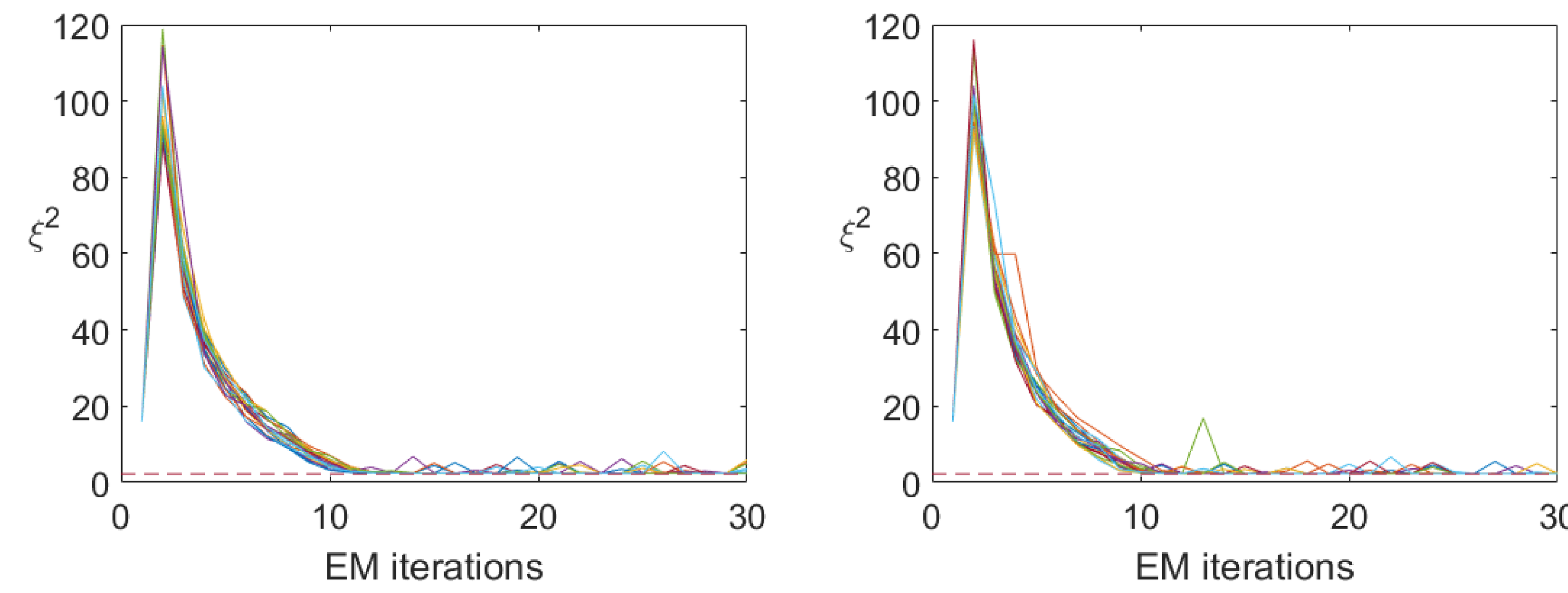

Figure 4: Convergence of estimations of ξ² for 20 runs of time length 500. SISR to the left and FLS with lag 32 to the right. The true values of ξ² is marked by the dotted red line.

n	Lag	0	2	4	8	16	32
100	$\hat{\eta}$	2.9960	2.9986	2.9989	2.9992	3.0003	3.0011
	std $\hat{\eta}$	0.0007	0.0008	0.0006	0.0010	0.0006	0.0006
	$\hat{\xi^2}$	2.6804	2.5392	2.3396	2.5712	2.5185	2.3476
	std $\hat{\xi^2}$	0.8552	0.6942	0.0124	0.6236	0.5667	0.0634
200	$\hat{\eta}$	3.0002	3.0013	3.0014	3.0013	3.0018	3.0015
	std $\hat{\eta}[10^{-3}]$	0.6648	0.3216	0.7251	0.3754	0.7752	0.7745
	$\hat{\xi^2}$	2.6319	2.5692	2.8468	2.6352	3.5642	2.6903
	std $\hat{\xi^2}$	0.6568	0.5366	1.2026	1.0467	3.0041	1.0875
500	$\hat{\eta}$	3.0020	3.0036	3.0037	3.0039	3.0038	3.0037
	std $\hat{\eta}[10^{-3}]$	0.3613	0.1823	0.2447	0.4382	0.4460	0.4187
	$\hat{\xi^2}$	2.5010	2.1764	2.2087	2.3762	2.2308	2.3028
	std $\hat{\xi^2}$	0.2765	0.0300	0.1463	0.4143	0.2177	0.4465

Table 2: Estimates and standard deviation of both η and ξ² for 100, 200 and 500 time steps. Both methods find reasonable estimates, however FLS estimates has notably less variance.

Conclusion

When comparing the estimated particle trajectories with the true trajectory, Figure 2, it is clear that the FLS trajectory indeed is much smoother than the one of the particle filter, which is very jagged by comparison. Using techniques such as FLS can be beneficial for applications with need of higher precision of the state estimates such as some tracking problems. However, if the interest is solely in estimating better parameters for a given model it might not be necessary to employ FLS in order to get reasonable results. This can be seen Table 2 where the difference in estimation between FLS and SISR is very small. Furthermore, in Figure 3-4 the difference in convergence speed between the two methods is minor.

References

- Jimmy Olsson et al. *Sequential Monte Carlo smoothing with application to parameter estimation in nonlinear state space models,* Bernoulli, 2008, Vol. 14, no. 1, 155-179.