Distributed High Accuracy Peer-to-Peer Localization in Mobile Multipath Environments

Venkatesan. N. Ekambaram and Kannan Ramchandran

Department of EECS, University of California, Berkeley

Email: {venkyne, kannanr}@eecs.berkeley.edu

Abstract—In this paper we consider the problem of high accuracy localization of mobile nodes in a multipath-rich environment where sub-meter accuracies are required. We employ a peer to peer framework where the vehicles/nodes can get pairwise multipath-degraded ranging estimates in local neighborhoods together with a fixed number of anchor nodes. The challenge is to overcome the multipath-barrier with redundancy in order to provide the desired accuracies especially under severe multipath conditions when the fraction of received signals corrupted by multipath is dominating. We invoke a message passing analytical framework based on *particle filtering* and reveal its high accuracy localization promise through simulations.

Key words- Localization, Multipath, Graphical Models, Particle Filtering, Hidden Markov Model.

I. INTRODUCTION

High-accuracy localization is mandated in many applications like vehicle safety [1], autonomous robotic systems [2], Unmanned Air Vehicle (UAV) systems etc, where sub-meter accuracies are called for. Standard GPS receivers can have errors over fifty or more meters which is unacceptable for many of these applications. The principal problem is multipath [1] interference [3], which is particularly prevalent in cities and "urban canyon" environments. In a multipath-rich environment, the received signals are no longer gaussian in nature challenging the use of standard estimation techniques like the well-known Kalman Filtering framework and its extensions. The principal idea behind GPS is to obtain three or more distance measurements from sources with known locations (e.g. satellites) and estimate the location based on trilateration. However, even if one of the measurements is corrupted by multipath, the location errors can be significantly large. It has been well noted that redundancy in measurements is the key to tackle multipath [3]. Many of the existing solutions such as D-GPS, A-GPS [4] augment the GPS system by adding extra infrastructure in terms of fixed base stations. However infrastructure cost and complexity constrains the amount of redundancy that can be introduced in the system and as a consequence limits the achievable localization.

In contrast to existing centralized systems that are based on a "cellular-like" architecture, with users individually computing their location by calculating the distance to a small number of satellites and/or terrestrial base stations, we adopt a "peer

---

[1]Multiple delayed versions of the same transmitted signal are received at the receiver due to reflections from different objects in the environment.
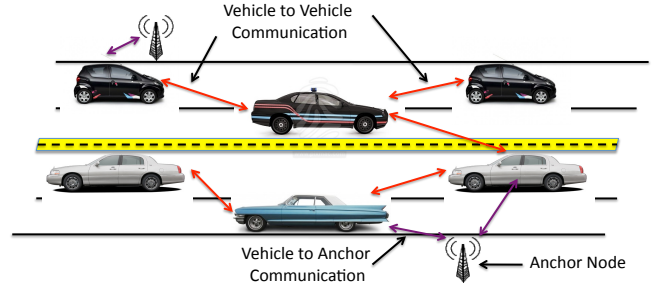

Figure 1. Peer-to-Peer Collaborative Localization

to peer" architecture (see Fig 1) where nodes collaborate and help each other to refine their position estimates. We look for distributed algorithms in the interest of scalability and reduced computational complexity. We propose the inclusion of low-cost static "anchor" nodes with known locations and study the effect of the number of anchor nodes and existing mobile nodes on the system performance. Collaboration coupled with mobility generates a large pool of measurements in the system. The fundamental insight is that, some fraction of these measurements will be produced by line-of-sight (LOS) dominated signals, and hence be fairly accurate, while some fraction will be corrupted by dominated non-line-of-sight (NLOS) reflected waves. Receivers do not know a priori which measurements are LOS and which are NLOS. Hence, the task of the users is to cooperatively discard the NLOS signals, thus enabling them to compute high-precision position estimates. Our main contribution in this paper is to uncover a framework and a distributed algorithm founded on *message passing* for collaborative localization. We adopt a mixture model (discussed in Sec. II) for the received signal to characterize LOS/NLOS, that naturally arises given that the receivers do not know a priori the nature of the received signal.

Distributed algorithms [2], [5] have been of recent interest for collaborative localization. However, most of the existing work in the literature on localization focus on the case when the measurements are gaussian. In our setting, the distributions of interest are mixture distributions that are highly non-gaussian in nature. Generalizations of Kalman Filtering, such as the Extended Kalman Filter (EKF) [6] have been proposed for non-gaussian problems. However, EKF solutions do not work well when the distributions are bimodal as in the case of mixture distributions, since gaussians do not approximate bimodal distributions very well. Some of the other approaches try to weed out the NLOS signals ([3], [7], [8], [9]), and work only with the LOS signals which can be reasonably modeled as gaussian. The underlying algorithm used in most

of these approaches has been modified forms of the Random Sample Consensus (RANSAC) algorithm, which is a classical algorithm in computer vision literature to discard outliers in the data. RANSAC works well when the number of outliers (here the NLOS signals) is much smaller than the number of LOS signals. However, these are limited in settings (for e.g. see [10] for an ultra-wideband setting) where the fraction of LOS signals is typically lesser than NLOS which would challenge the performance of RANSAC-based approaches. The main reason for this is that in our case the NLOS signals dominate the set of measurements, and those can no longer be considered as outliers. Simulation results show that our algorithm works very well even in this challenging case where the fraction of LOS signals is dominated by the fraction of NLOS signals. An excellent survey of existing locationing algorithms and their drawbacks can be found in [11].

II. PROBLEM SETUP

We have N mobile nodes with an arbitrary mobility model and M static anchor nodes with known locations. For simulations we consider the nodes to be moving with a constant velocity along a fixed trajectory, motivated by the highway setting for vehicles. Each vehicle is equipped with a sensor capable of getting a Time of Arrival (ToA)/ Time Difference of Arrival (TDoA) signal from other vehicles/anchors within a communication radius R. We model each measurement as either a LOS-dominated signal or an NLOS-dominated signal by choosing the observation noise in the received signal to be drawn from a mixture of two distributions, corresponding to LOS and NLOS respectively. The model is motivated by some of the experimental work carried out in the UWB [10], [12], [13]. For e.g., the experimental results in [10] show that some fraction of the received signals are purely LOS-dominated signals which motivates the mixture distribution. For simulations, we will model the noise in LOS as gaussian whereas for the NLOS we will take it to be a sum of an exponential and a gaussian distribution. The noise model for NLOS is again motivated from some of the experimental results discussed in [12], [13]. These show that the NLOS signal distribution is very close to an exponential distribution. This was earlier conjectured in [14], where the author argues that the arrivals of the different paths can be modeled as a Poisson process which in turn leads to the inter arrival times being exponential. The theory developed here is however, more generic and does not depend on the specific nature of the distributions. Each vehicle is assumed to be equipped with an accelerometer and magnetometer that give noisy readings of the velocity and direction of motion of the vehicle. We will assume that there is a multiple access protocol in place that will help the vehicles communicate across the shared medium.

Let $v_t(k)$ denote the true location of the kth vehicle at time instant t. $\theta_t(km)$ denotes the reading (distance measurement) between the vehicle k and node[2] m at time instant t. The readings $\theta_t(km)$

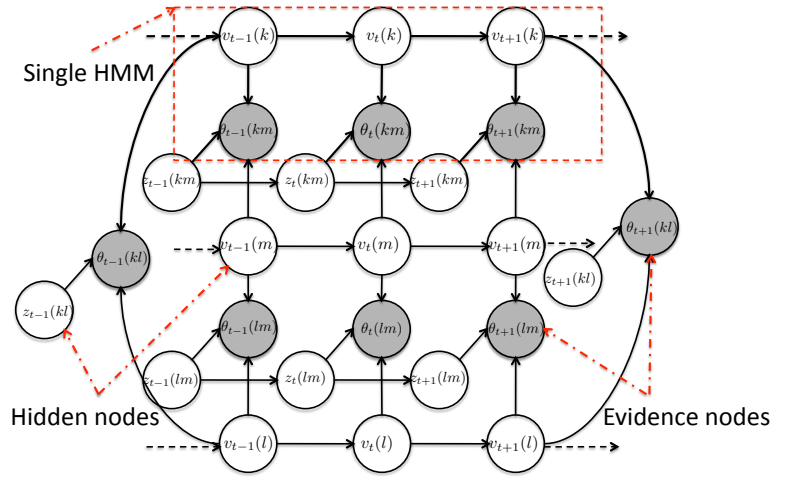[2]The term vehicle/anchor and node will be interchangeably used.

Figure 2. Graphical model representation of the unknown vehicle states (locations) and the observations (LOS/NLOS measurements).

are sampled from a mixture of two distributions, $(p_{LOS}(\theta_t(km)|v_t(k), v_t(m)), p_{NLOS}(\theta_t(km)|v_t(k), v_t(m))))$, with mixture probabilities $(\alpha, 1-\alpha)$ respectively. Let $z_t(km)$ be the indicator random variable for the LOS reading between vehicles k and m at time instant t.

$$z_t(km) = \begin{cases} 1 & \text{if } \theta_t(km) \sim p_{LOS}(\theta_t(km)|v_t(k), v_t(m)) \\ 0 & \text{otherwise} \end{cases}$$

It is highly unlikely that a vehicle's state would change rapidly over time. To capture this, we will model the state evolutions of the $z_t(km)$'s as a "sticky" Markov Chain with a stationary distribution $(\alpha, 1-\alpha)$. Let $p(v_t(k)|v_{t-1}(k))$ be the distribution that governs the evolution of the vehicle states across time which is an artifact of the inertial navigation system.

Every vehicle k needs to estimate its location $v_t(k)$, based on all measurements $\{\theta_\tau(km)\}_{\tau=1}^t$ from its neighbors upto time t. Given the non-gaussian nature of the problem and the accuracy requirement we need to develop robust, accurate and distributed algorithms. *Particle Filters* have gained importance in the recent past for tackling non-gaussian estimation problems. These provide accuracies close to Minimum Mean Square Error (MMSE) estimates. The nature of our problem helps us obtain Kalman-like updates for particle filtering giving rise to a simplified algorithm. A short primer on graphical models and particle filtering is provided in the next section.

III. GRAPHICAL MODELS & PARTICLE FILTERING PRIMER

Graphical models and particle filtering have been extensively studied in the machine learning community. An excellent treatment of graphical models is provided in [15] and discussions on particle filtering can be found in [16]. We describe these briefly to make our paper self contained.

A. Graphical models

Graphical models provide a good way of understanding dependencies between different random variables. These models come in handy when we are trying to estimate some parameters given probability distributions that have a certain structure and we need computationally feasible algorithms to get the estimates. A directed acyclic graph, $\mathcal{G}(\mathcal{V}, \mathcal{E})$, consists of a vertex set $\mathcal{V}$ and edge set $\mathcal{E}$ which is the collection of all directed edges. The vertices (nodes) of the directed graph

Algorithm 1 Particle Filtering

1: **Initialize:** Sample $\{v_0^i\}_{i=1}^K$ from $p(v_0)$, set $w_0^i = \frac{1}{K}$.
2: **for** $t \geq 1$ **do**
3:     Sample v_t^i from $p(v_t|v_{t-1}^i)$.
4:     Update $w_t^i = w_{t-1}^i p(\theta_t|v_t^i)$.
5:     Normalize w_t^i.
6:     Resample new set of particles and set $w_t^i = \frac{1}{K}$
7: **end for**
8: **Output:** $\hat{E}[f(v_t)|\theta_{1:t}] = \sum_{i=1}^K w_t^i f(v_t^i)$.

Figure 3. Approximation for the original coupled HMM.

represent random variables and the dependencies amongst the random variables are captured by the edges. Let $\{X_s, s \in \mathcal{V}\}$ denote the set of all random variables indexed by the nodes of the graph. For every node $s \in \mathcal{V}$, let π_s denote the set of indices of its parents [3]. For any $S \subseteq \mathcal{V}$, let $X_S \triangleq \{X_s, s \in S\}$. Then, the joint probability distribution of the random variables can be factored as $p(X_\mathcal{V}) = \prod_{s \in \mathcal{V}} p(X_s|X_{\pi_s})$.

Dependencies amongst the vehicle locations and the readings are captured by the graphical model shown in Fig 2, which is a coupled Hidden Markov Model (HMM). The unshaded nodes are the *hidden nodes* to be estimated [4] and the shaded nodes are observations coupling the different Markov chains of the vehicles, called as the *evidence nodes*. Based on the factorization described above, the joint probability distribution of the set of all random variables is given by equation (1).

The primary goal here is to estimate the hidden nodes given the observations. It is hard to directly apply conventional inference algorithms such as the celebrated loopy belief propagation [15] considering the fact that the hidden states are continuous in nature and that the graph grows over time. We will introduce approximations to the graphical model and use particle filtering to do inference over the approximated model. Simulation results show that we can obtain high accuracies in spite of the approximations. We briefly describe particle filters in the next section.

B. Particle Filtering

Particle Filtering is a Monte Carlo simulation technique whose goal is to approximate the posterior state density that can be used to obtain the MMSE estimate. We will consider only a single vehicle for now and omit the vehicle index k for notational simplicity. Let $v_{1:t}$ and $\theta_{1:t}$ denote the set of states and observations up to time t respectively. The posterior density of the state given the observations can be approximated as $p(v_t|\theta_{1:t}) \approx \hat{p}(v_t|\theta_{1:t}) = \frac{1}{K}\sum_{i=1}^K \delta(v_t - v_t^i)$ where $\{v_t^i; i = 1, ..., K\}$ are K i.i.d random samples (particles) picked from the distribution $p(v_t|\theta_{1:t})$ and $\delta(.)$ is the dirac-delta function. Expectations of some functions of the state $f(v_t)$, can then be approximated with high probability, for large K, as $E[f(v_t)|\theta_{1:t}] \approx \frac{1}{K}\sum_{i=1}^K f(v_t^i)$. Typically sampling from the true posterior distribution is hard. Let us consider sampling from a distribution $\pi(v_{1:t}|\theta_{1:t})$, known as *proposal distribution*. After some simple manipulations using

[3]Node p is the parent of node s, if there is an incoming edge to s from p.
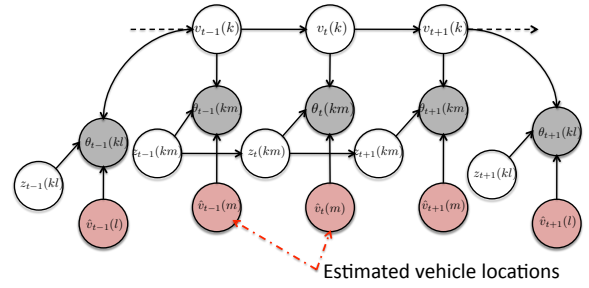[4]Anchor nodes are not shown in this model for simplicity.

Bayes rule, we get

$$E[f(v_{1:t})|\theta_{1:t}] = \frac{E_\pi[w_t(v_{1:t})f(v_{1:t})]}{E_\pi[w_t(v_{1:t})]},$$

where $w_t(v_{1:t})$ are known as the importance weights given by

$$w_t(v_{1:t}) = \frac{p(\theta_{1:t}|v_{1:t})p(v_{1:t})}{\pi(v_{1:t}|\theta_{1:t})}.$$

Thus, if we have i.i.d samples of $v_{1:t}$ from the proposal distribution $\pi(v_{1:t}|\theta_{1:t})$, we get $E[f(v_{1:t})|\theta_{1:t}] \approx \sum_{i=1}^K \tilde{w}_t^i f(v_{1:t}^i)$, where $\tilde{w}_t^i$ are the normalized importance weights. Convergence of this estimate to the true MMSE estimate is shown in [16].

For the case of a single HMM (Fig 2), we have $p(v_{1:t}, \theta_{1:t}) = p(v_1) \prod_{j=2}^t p(v_j|v_{j-1}) \prod_{j=1}^t p(\theta_j|v_j)$. Lets choose a proposal distribution that admits a decomposition $\pi(v_{1:t}|\theta_{1:t}) = \pi(v_{1:t-1}|\theta_{1:t-1})\pi(v_t|v_{1:t-1}\theta_{1:t})$. After simplifications, we can obtain a recursive estimation for the weights,

$$w_t = \frac{w_{t-1}p(\theta_t|v_t)p(v_t|v_{t-1})}{\pi(v_t|v_{1:t-1}\theta_{1:t})}.$$

The optimal proposal distribution, that minimizes the variance of the error has been shown [16] to be $p(v_t|v_{t-1}, \theta_t)$. Since sampling from this distribution is difficult for our problem, we will use a simpler distribution $\pi(v_t|v_{1:t-1}\theta_{1:t}) = p(v_t|v_{t-1})$ to get the weight updations $w_t = w_{t-1}p(\theta_t|v_t)$. The reader is referred to the literature for other methods of choosing a proposal distribution. Typically a resampling step is introduced (see Alg. 1), to handle degeneracy issues when particle weights become too low.

IV. PARTICLE FILTERING FOR LOCALIZATION

Exact inference over Fig 2 being hard, we will resort to an approximation for every vehicle as shown in Fig 3. The new set of shaded nodes in this graph correspond to the estimated location of the other vehicles. At every time instant each vehicle gets the estimated location of its neighbors from the previous time instants and assuming that it is close enough to the true location, the vehicle gets an estimate of its own location using particle filtering. A straightforward method of particle filtering over this model would be to consider $\{v_t, z_t\}$ as a random variable pair which would reduce the graphical model to a

$$p(\{v_t\}, \{\theta_t\}, \{z_t\}) = \prod_{t,k,m} p(\theta_t(km)|v_t(k), v_t(m), z_t(km)) \prod_{k=1}^N p(v_1(k)) \prod_t p(v_t(k)|v_{t-1}(k)) \prod_{k,m} p(z_1(km)) \prod_t p(z_t(km)|z_{t-1}(km)). \quad (1)$$

Algorithm 2 Particle Filtering for Accurate Localization

1: **Initialize:** Sample $\{v_0^i(k)\}_{i=1}^K$ from $p(v_0(k))$, set $w_0^i(k) = \frac{1}{K}$ $\forall k$.
2: **for** $t \geq 1$ **do**
3:    Sample $v_t^i(k)$ from $p(v_t(k)|v_{t-1}^i(k))$.
4:    **for all** Neighbors m **do**
5:      **if** m was a neighbor at $t-1$ **then**
6:       $\phi_t^i(km) = p(\theta_t(km)|z_t(km), v_t^i(k))$
        $\sum_{z_{t-1}} p(z_t(km)|z_{t-1}(km))\phi_{t-1}^i(km)$.
7:      **else**
8:       $\phi_t^i(km) = p(z_t(km))p(\theta_t(km)|v_1^i(k), z_t(km))$.
9:      **end if**
10:    **end for**
11:    Calculate $w_t^i(k) = \prod_m \sum_{z_t(km)} \phi_t^i(km)$
12:    Normalize w_t^i and resample.
13: **end for**
14: **Output:** $\hat{E}[v_t(k)|\{\theta_{1:t}(km)\}] = \sum_{i=1}^N w_t^i(km)v_t^i(k)$.

simple HMM. However the state space of the random variables grows exponentially with the number of neighbors as we now need to sample particles over $\{v_t(k), z_t(km)\}$, which could lead to scaling issues. More importantly particle filtering is efficient when the hidden states are continuous, whereas z_t's are binary. Thus we now see to combine particle filtering and exact inference to obtain simplified weight updations.

Consider the HMM in Fig 3 and ignore the vehicle indices k and m. Using the same definition of $w_t(v_{1:t})$ as before, we can now write $w_t(v_{1:t}) = \sum_{z_t} \phi(v_{1:t}, z_t)$, where

$$\phi(v_{1:t}, z_t) = \frac{p(\theta_{1:t}, z_t|v_{1:t})p(v_{1:t})}{\pi(v_{1:t}|\theta_{1:t})}.$$

Assuming a similar proposal distribution factorization as before we can show that

$$\phi(v_{1:t}, z_t) = \frac{p(\theta_t|z_t, v_t)p(v_t|v_{t-1})}{\pi(v_t|v_{1:t-1}\theta_{1:t})} \sum_{z_{t-1}} p(z_t|z_{t-1})\phi(v_{1:t-1}, z_{t-1}).$$

Choosing $\pi(v_t|v_{1:t-1}\theta_{1:t}) = p(v_t|v_{t-1})$, we get

$$\phi_t \triangleq \phi(v_{1:t}, z_t) = p(\theta_t|z_t, v_t) \sum_{z_{t-1}} p(z_t|z_{t-1})\phi(v_{1:t-1}, z_{t-1}).$$

For each vehicle k and its neighbor m we have $\phi_t(km)$ and the update equations are given in the Algorithm 2.

The optimal detection rule for z_t is given by

$$\hat{z}_t = \begin{cases} 1 & \text{if } p(z_t = 1|\theta_{1:t}) > p(z_t = 0|\theta_{1:t}) \\ 0 & \text{otherwise} \end{cases}.$$

This can be simplified and shown to be equal to the following test $E_\pi \phi(v_{1:t}, z_t = 1) \gtrless_0^1 E_\pi \phi(v_{1:t}, z_t = 0)$, which is evaluated by approximating $E_\pi \phi(v_{1:t}, z_t) \approx \sum_i \phi(v_{1:t}^i, z_t)$. To take care of degeneracy issues over long time instants [16], we reset the system when the number of distinct particles become small and sample a new set of particles from a neighborhood around the location estimate at that time instant.

V. SIMULATION RESULTS

The simulation set up consists of $N = 20$ vehicles moving in a grid of size $150m \times 30m$. The vehicles start at random locations from the left most corner of the grid and move at a constant velocity to the right. To account for the vertical motion, a simplistic model of a curved trajectory is simulated for each vehicle. The observations are generated as follows. If $d_t(km) = ||v_t(k) - v_t(m)|| < R$,

$$\theta_t(km) = \begin{cases} d_t(km) + n_t(km) & \text{if LOS} \\ d_t(km) + \epsilon_t(km) + n_t(km) & \text{otherwise} \end{cases}$$
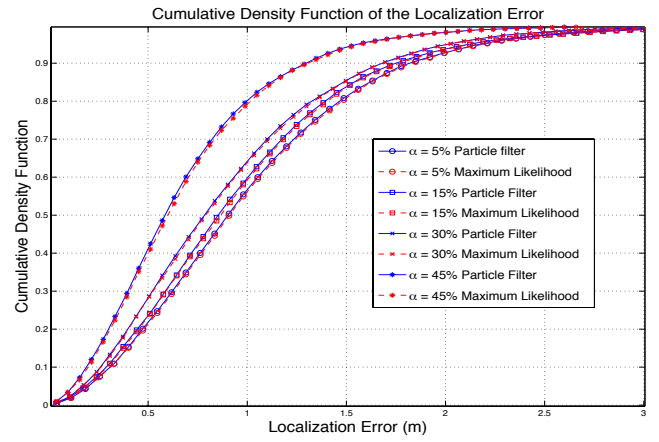
(a) Cumulative density function of localization error as a function of α.
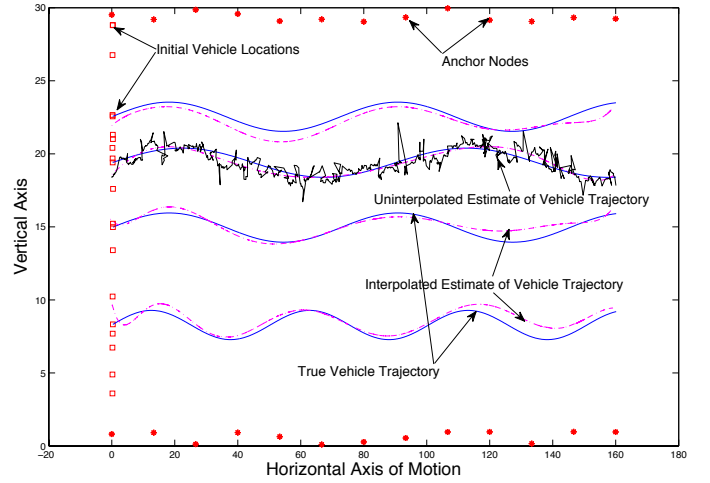
(b) True and estimated vehicle trajectories for the proposed algorithm, $\alpha = 15\%$

α (%)	P_d	Mean Error (Particle Filtering)	Mean Error (Maximum Likelihood)
5	0.43	1.04 m	1.05 m
15	0.52	0.99 m	1.01 m
30	0.61	0.93 m	0.94m
45	0.68	0.71 m	0.73 m

(c) Mean estimation errors and the detection probability of the LOS signals (P_d) as a function of α.

Num of Anchors (M)	Num of Vehicles (N)	Mean Error (Particle Filtering)
26	20	0.93 m
26	24	0.82 m
26	30	0.76 m
30	20	0.74 m
36	20	0.70 m

(d) Mean estimation errors of the particle filtering algorithm as a function of the number of anchor nodes and vehicles for $\alpha = 30\%$.

Figure 4. Parameters for all the plots: $\sigma_{INS} = 0.1$, $\sigma_{LOS} = 0.05$, $\sigma_{NLOS} = 5$, Num of particles = 900, Horizontal velocity= $0.2m$ per time step $\forall k$, $M = 26$, $N = 20$, $R = 10m$. The plots are a function of the fraction of LOS signals α. The proposed particle filtering algorithm is compared against the local Maximum Likelihood algorithm.

where $n_t(km) \sim N(0, \sigma_{LOS}^2)$ i.i.d, $\epsilon_t(km) \sim Exp(\sigma_{NLOS}^{-1})$. We assume the fraction of the readings that are LOS to be α. The evolution of the $z_t(km)$ random variables is governed by the probability law, $p(z_t(km) = 1|z_{t-1}(km) = 0) = \frac{\alpha}{2}$ for these simulations and the other values are taken so that the stationary distribution is $(\alpha, 1-\alpha)$. The inertial navigation

system readings are assumed to be obtained under an additive white gaussian noise model. Time steps are divided into units of one for simplicity. We compare the performance of the algorithm to a local genie aided Maximum Likelihood (ML) algorithm. Here each vehicle, at every time instant, calculates the local ML estimate of its location assuming that a genie provides it with the exact locations of its neighbors.

The localization error cumulative density function is plotted for different values of α in Fig 4(a). The x-axis is the set of error values and y-axis is the cumulative density function. One can see that even at low values of $\alpha = 5\%$, more than 80% of the errors are less than $1.5m$. The algorithm performance is slightly better than ML algorithm. This is not too surprising considering that particle filtering tries to approximate MMSE which is the optimal solution for mean squared loss function. Algorithms based on RANSAC were found to have errors over $5m$ and are not discussed here. The true and estimated vehicle trajectories are plotted in Fig 4(b). The estimated vehicle trajectory is generated by a simple polynomial fit to all the estimated vehicle locations. Table 4(c) shows the probability of detection (P_d) of the hidden $z_t(km)$ states (fraction of the times LOS states are detected correctly) and the mean estimation error. The strength of the algorithm is in exploiting the large pool of measurements efficiently where algorithms like RANSAC fail. Table 4(d) shows the localization error performance as a function of the number of anchor nodes and mobile nodes in the system. One can observe a law of diminishing returns as the number of anchor nodes increases. Another interesting observation is that the percentage improvement in the localization performance when the number of mobile nodes, that act like "pseudo-anchors", is the similar to that of adding more anchors. A theoretical exploration of this phenomenon is currently under progress.

We shall now discuss some practical aspects of implementing this scheme. The communication between the nodes can be carried over the Dedicated Short Range Communications (DSRC) band, that has been allocated by the Federal Communications Commission (FCC) for Intelligent Transportation Systems. IEEE 802.11p can be used for medium access. This is essentially a CSMA protocol specialized for vehicular networks. Whenever a node receives a 802.11p packet from some other node, it can estimate the time delay of arrival that translates to a distance measurement used for localization. The location estimate sent by one node to the other node can be sent as part of the 802.11p packet. The parameters of the distributions and the fraction α can be estimated using an Expectation Maximization (EM) algorithm at every step. Each vehicle has an average of 8 other vehicles and 4 anchor nodes in its communication radius which is quite a reasonable assumption. In practice, the anchors usually have a larger range of communication and the anchor density required would be lesser than that used in the simulations. We assume that the computational load in updating the particles is manageable by present day vehicles that have a reasonably high processing power.

VI. Conclusion and Future work

In this work we explored the application of particle filtering to get estimates of vehicle locations in a highly NLOS environment. We derived weight update equations for the NLOS setting and simulation results show that reasonably good accuracies in positioning is feasible. Future work includes carrying out more realistic simulations using traffic and network simulators. The approximation in the graphical model could break down above a certain noise threshold and below a certain anchor density, and the algorithm could potentially diverge. A theoretical understanding of when the algorithm diverges is another research direction, though we believe this to be a hard problem. A theoretical exploration of the effect of increased anchor and mobile nodes is under progress. Integrating other sensing modalities into the algorithm is a future research direction.

References

[1] C. Wilson, S. Rogers, and S. Weisenburger, "The potential of precision maps in intelligent vehicles," in *IEEE International Conference on Intelligent Vehicles*. Citeseer, 1998, pp. 419–422.

[2] E. Nerurkar, S. Roumeliotis, and A. Martinelli, "Distributed maximum a posteriori estimation for multi-robot cooperative localization," in *IEEE International Conference on Robotics and Automation, 2009. ICRA'09*, 2009, pp. 1402–1409.

[3] P. Chen, "A non-line-of-sight error mitigation algorithm in location-estimation," in *1999 IEEE Wireless Communications and Networking Conference, 1999. WCNC*, 1999, pp. 316–320.

[4] G. Djuknic and R. Richton, "Geolocation and assisted GPS," *Computer*, vol. 34, no. 2, pp. 123–125, 2001.

[5] N. Patwari, J. Ash, S. Kyperountas, A. Hero III, R. Moses, and N. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54–69, 2005.

[6] E. Kiriy and M. Buehler, "Three-state extended kalman filter for mobile robot localization," *McGill University., Montreal, Canada, Tech. Rep. TR-CIM*, vol. 5, 2002.

[7] B. Ananthasubramaniam and U. Madhow, "Cooperative localization using angle of arrival measurements in non-line-of-sight environments," in *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*. ACM, 2008, pp. 117–122.

[8] R. Casas, A. Marco, J. Guerrero, and J. Falcó, "Robust estimator for non-line-of-sight error mitigation in indoor localization," *EURASIP Journal on Applied Signal Processing*, vol. 17, 2006.

[9] C. Seow and S. Tan, "Non-Line-of-Sight Localization in Multipath Environments," *IEEE Transactions on Mobile Computing*, pp. 647–660, 2008.

[10] M. Renzo, F. Graziosi, R. Minutolo, M. Montanari, and F. Santucci, "The ultra-wide bandwidth outdoor channel: from measurement campaign to statistical modelling," *Mobile Networks and Applications*, vol. 11, no. 4, pp. 451–467, 2006.

[11] A. Boukerche, H. Oliveira, E. Nakamura, and A. Loureiro, "Vehicular ad hoc networks: a new challenge for localization-based systems," *Computer Communications*, vol. 31, no. 12, pp. 2838–2849, 2008.

[12] G. Turin, F. Clapp, T. Johnston, S. Fine, and D. Lavry, "A statistical model of urban multipath propagation," *IEEE Transactions on Vehicular Technology*, vol. 21, no. 1, pp. 1–9, 1972.

[13] K. Pedersen, P. Mogensen, and B. Fleury, "A stochastic model of the temporal and azimuthal dispersion seen at the base station in outdoor propagation environments," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 2, pp. 437–447, 2000.

[14] G. Turin, "Communication through noisy, random-multipath channels," Ph.D. dissertation, Massachusetts Institute of Technology, 1956.

[15] M. Jordan, *Learning in graphical models*. Kluwer Academic Publishers, 1998.

[16] A. Doucet, N. De Freitas, and N. Gordon, *Sequential Monte Carlo methods in practice*. Springer Verlag, 2001.