This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.

Cooperative Vehicle Position Estimation

Ryan Parker and Shahrokh Valaee
The Edward S. Rogers Sr. Department of Electrical and Computer Engineering
University of Toronto, 10 King's College Road, Toronto, ON, M5S 3G4, Canada
e-mail:{rparker,valaee}@comm.utoronto.ca

***Abstract*— We present a novel cooperative vehicle position estimation algorithm, which can achieve higher levels of accuracy and reliability than existing GPS based positioning solutions by making use of inter-vehicle distance measurements taken by a radio ranging technology. Our algorithm uses signal strength based inter-vehicle distance measurements, road maps, vehicle kinematics, and Extended Kalman Filtering to estimate relative positions of vehicles in a cluster. We have preformed analysis of our algorithm examining its performance bounds, computational complexity and communication overhead requirements. Also, we have shown that the accuracy of our algorithm is superior to previous proposed localization algorithms.**[1]

I. Introduction

Recent advances in intelligent transportation systems (ITS), with the advent of the 802.11p based dedicated short range communication (DSRC) devices, has opened the door for a new era of wireless applications. The 802.11p standard for Wireless Access in Vehicular Environments (WAVE) is designed to support short range, low latency, high speed vehicle-to-vehicle and vehicle-to-infrastructure (e.g. road side access points) wireless communication. Currently, DSRC devices are used for electronic toll collection systems. However, in the future, these devices will enable an enhanced level of safety, efficiency and information availability, by using vehicle-to-vehicle and vehicle-to-roadside communication to provide real-time information about hazards that lie on the road ahead (e.g. road construction) [1].

Recently, a cooperative collision warning system based on inter-vehicle communications has been receiving considerable interest among researchers, government and industry. This cooperative collision warning system will work by vehicles cooperatively sharing information (i.e. location, speed, heading, acceleration, etc.), via DSRC, for collision anticipation. It was shown in [2] that by sending safety warning messages the probability of collision in a platoon of vehicles can be substantially reduced. However, in order to enable the operation of such a system, it is required that a vehicle build a map of the relative location of neighbouring vehicles, in an accurate and reliable way.

Currently, the dominant technology for determining a vehicle's position is the global positioning system (GPS). Regular GPS can provide an accuracy of 10 meters when there is direct line-of-site between the vehicle and four or more satellites [3]. The 10 meter accuracy of regular GPS can be improved by using differential GPS (DGPS), which can achieve accuracies between 3 and 7 meters [3]. However, the GPS signals can often become blocked or degraded while a vehicle is traveling under bridges, through tunnels, or in downtown areas among tall skyscrapers resulting in inaccurate position estimates. To combat these gaps in the availability of the GPS signal, vehicles can make use of dead-reckoning systems to maintain an estimate of their position, using onboard kinematic sensors [3]. However, dead-reckoning systems are prone to error accumulation. For example, if the GPS signal is lost for 30 seconds the position estimate can become inaccurate by as much as 10-20 meters, for a vehicle traveling at 100km/h.

In this paper, we show that gains in accuracy and reliability can be achieved over GPS-based approaches by making use of inter-vehicle distance measurements taken by a radio based ranging technology. Also, that the accuracy of previously proposed radio ranging based localization method can be improved by taking into account information such as road maps and vehicle kinematics. We will propose a Kalman filter based solution and show that our algorithm is accurate and reliable, and allows a real-time implementation. Also, we will show that the structure of our algorithm allows us to place a probabilistic bound on the confidence in the position estimate, so that inaccurate position estimates can be actively identified.

Before exploring the details of our algorithm we will, in the next section, provide some background information on the problem we are trying to solve as well as discuss some previous works related to ours. After this, we will present the details of the structure algorithm works. Then present the results of series of experiments, where we compare our algorithm to two previously proposed algorithms. Following this, we will provide a detailed analysis of our algorithm in terms of its performance bounds and computational complexity. Then, lastly finish the paper with our conclusions.

II. Related Work and Background Information

The idea of using a radio based ranging technology for node positioning is not new. In the last few years, this problem has been tackled by researchers for stationary sensor networks (e.g. [4], [5], [6]). However, the problem of vehicle localization is different as the nodes (vehicles) in the network are all mobile. Previously proposed stationary localization approaches offer no straightforward extension to the mobile case. In [5], it was shown that mobility makes localization much more difficult and that position estimation errors increase with speed. Also, in localization of sensor

[1]This research has been supported in part by in-kind contributions from Mark IV industries.

1-4244-0353-7/07/$25.00 ©2007 IEEE

nodes, generally, the goal is to determine the position of each sensor with respect to a global coordinate system, under the assumption that a subset of the sensor nodes in the network have prior prefect knowledge of their location. For our case, we wish to only establish the relative positions of the vehicles. This can be seen as a more generalized version of the problem for sensor networks, since a localization solution based on inter-vehicle distances alone is valid for an arbitrary translation, rotation and reflection of the network. Therefore, for fairness of comparison we will choose to compare our method to other previously proposed approaches that estimate relative node positions. In [6], Moore et al. present *robust quads* algorithm that allows each node to derive the relative positions of its neighbours. Unfortunately, the algorithm presented in [6] achieves its level of robustness by localizing only those nodes that have a high probability of having an unambiguous position (i.e. will not necessarily localize all nodes in a cluster).

Other algorithms for relative position estimation are based on multi-dimensional scaling (MDS), such as the ones presented in [7] and the references therein. In general, the MDS approach solves a robust weighted non-linear least squares optimization problem. The majority of the works only consider static localization. However, there have been extensions proposed for the nonlinear least square problem to include velocity and lane constraints in the localization problem [8]. The work in [8] was later reformulated, so that the maximum error in the position estimates was minimized [9]. Both [8] and [9] place hard bounds on the possible positions of the vehicles in the cluster. The hard bounds are found by introducing a noise margin in the location estimates. However, this noise is difficult to be described at each time step. In this paper, we incorporate the noise in the velocity readings by describing the probability distribution of the noise. In the case of Gaussian noise, all that is required is the mean and variance. Overall, this has the effect of reducing the algorithm sensitivity where the error in one of the inter-vehicle distance measurements is relatively high. Also, it allows us to structure our algorithm so that instead of the algorithm returning a set of position estimates, as in [8] and [9], we can assign a probabilistic level of uncertainty to each position estimate.

In [10], the mobility for the sensor networks has been used to improve the accuracy of the location estimator. However, they have assumed that at each time step each node is able to move in an arbitrary direction, with a bound on the maximum speed of the movement. In our case, for a vehicular network, we know that vehicles travel on roads, and in general within lanes. Also, it is possible to determine the velocity of the vehicle. Both of which provide a different set of constraints for the localization problem.

Given these different approaches the two closest to our work is the robust quads algorithm presented by Moore et al. in [6] and the vehicle based nonlinear least squares approach presented in [8]. Therefore, we will compare the algorithm presented in this paper to these methods.

A. Radio Based Ranging Techniques

In general, localization schemes discussed in the previous section are independent of the ranging technology used for distance estimates — all they require is that distance estimates are obtained by some means. However, for completeness we will mention that distance estimates are commonly found using received signal strength indicator, time-of-flight, or angle-of-arrival. Each of these techniques has been studied extensively in literature. We refer the reader to [11] for a good overview of these techniques and [12] for an analysis of the performance limitations of each of these techniques.

III. Problem Definition

The vehicle position estimation problem can be formulated as follows. Consider a cluster of n vehicles labeled 1,2,...n at unknown distinct locations in some physical region at time t. For each vehicle, we establish a map of the relative position of its neighbours. More explicitly, we estimate the true relative positions of the vehicles, **A**, defined as:

$$\boldsymbol{A} = [x_1, x_2...x_n, y_1..y_n] \tag{1}$$

where $\{x_i, y_i\}_{i=1}^n$ represents the relative position of the ith vehicle in a coordinate system with the location of the vehicle performing the mapping set at the origin $(0, 0)$.

These position estimates will be created based on three main factors: inter-vehicle distance estimates, made by using a radio based ranging technology; velocity information from each vehicle, derived from onboard sensors; and using a road map to ensure that position estimates are within the road boundaries.

A two step process is performed to gain information about the inter-vehicle distances and velocities of all vehicles in the cluster. In the first step, inter-vehicle distance measurements are made by each vehicle using a radio ranging technology to estimate their relative distance. Also, in the first step vehicles within the cluster will read their own speed information. In the second step, the information each vehicle collected is shared with its neighbours, which can be accomplished by standard multicast techniques. So, after the second step each vehicle has up to $n \times (n-1)$ inter-vehicle distance readings and a set of n velocity readings. Note, that the distance measured from vehicle i to vehicle j, may not be equal to the distance measured from j to i.

IV. Algorithm Framework

A. Extended Kalman Filter Approach

We use Kalman filtering since it allows a recursive set of operations that produces an estimate of the positions (state) by processing data from the inter-vehicle distance estimates (observations) and incorporates this into a motion model. Also, the Kalman filter provides the optimal set of position estimates in the mean square sense, under the assumption of Gaussian noise distribution for the mobility and measurement model [13].

The motion model we will use to incorporate the velocity measurements (also often referred to as the state equation) is:

$$\boldsymbol{A}_k = \boldsymbol{A}_{k-1} + T_s \mathbf{u}_{k-1} + T_s \mathbf{w}_{k-1} \tag{2}$$

where

$$\boldsymbol{A}_k = [x_{1,k}, x_{2,k} ... x_{n,k}, y_{1,k} ... y_{n,k}]^T \tag{3a}$$

$$\mathbf{u}_{k-1} = [v_{x1,k-1}, ... v_{xn,k-1}, v_{y1,k-1} ... v_{yn,k-1}]^T \tag{3b}$$

where n is the number of vehicles in the cluster at the kth snap shot; T_s is the sampling interval; superscript T denotes transposition, $v_{xi,k-1}, v_{yi,k-1}$ is the velocity of vehicle i in the x and y directions at time $k-1$, respectively; $\boldsymbol{A}_k$ is the position of vehicle at time k; and $\mathbf{w}_{k-1}$ is the process noise describing the mobility variations. We assume that $\mathbf{w}_{k-1}$ is a zero-mean Gaussian random variable, with the following covariance:

$$\mathbf{Q}_{k-1} \triangleq \mathrm{E}\{\mathbf{w}_{k-1}\mathbf{w}_{k-1}^T\} = \mathrm{diag}(\sigma_{x1}^2 .... \sigma_{xn}^2, \sigma_{y1}^2 ... \sigma_{yn}^2) \tag{4}$$

where $\mathrm{diag}(\cdot)$ denotes a diagonal matrix.

Considering that vehicles usually move along lanes in roads, the uncertainty in the direction of road is greater than the uncertainty in the direction orthogonal to the road. Let us define the variance in direction of the road for vehicle i to be $\sigma_{i,a}^2$ and the variance in the direction orthogonal to be $\sigma_{i,o}^2$. Due to the higher uncertainty of the motion along the road $\sigma_{i,a}^2 \gg \sigma_{i,o}^2$. Thus, for a road that runs in a direction that is θ radians from the y axis, the following transformation must be applied:

$$\sigma_{xi}^2 = \sigma_{i,o}^2 \cos^2\theta + \sigma_{i,a}^2 \sin^2\theta \tag{5}$$

$$\sigma_{yi}^2 = \sigma_{i,o}^2 \sin^2\theta + \sigma_{i,a}^2 \cos^2\theta \tag{6}$$

This allows $\mathbf{Q}_{k-1}$ to be biased in the direction of the road.

The observations of the inter-vehicle measurements are expressed as:

$$\mathbf{z}_k = \mathbf{h}_k(\boldsymbol{A}) + \mathbf{v}_k \tag{7}$$

where $\mathbf{h}_k(\boldsymbol{A})$ is a nonlinear equation describing the measurements at time k and $\mathbf{v}_k$ is a zero mean Gaussian random vector, with the covariance matrix $\mathbf{R}_k$, describing the noise characteristics of the measurements. The measurement equation is nonlinear, since the distance between vehicles i and j equals $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$. Note, that the general form of the Kalman filter requires that the measurement (observation) equation be in a linear form, therefore we will linearize (7) using the first-order Taylor series expansion around the current position estimates. Let us define

$$\hat{\mathbf{H}}_k = \left. \frac{d\mathbf{h}_k(\boldsymbol{A})}{d\boldsymbol{A}} \right|_{\boldsymbol{A}=\mathbf{A}_{k|k-1}} \tag{8}$$

where $\hat{\mathbf{H}}_k$ can be referred to as the Jacobian matrix and $\mathbf{A}_{k|k-1}$ the estimate of $\mathbf{A}_k$ using the observation up to time $k-1$. Given (2) and the nonlinear form of (7), we can form the extended Kalman filter (EKF) algorithm. The size of the $\hat{\mathbf{H}}_k$ for a cluster of n vehicles is $(n^2-n)\times(n-1)$. To reduce the computational complexity we can reduce the number of rows in the $\hat{\mathbf{H}}_k$ matrix to $(n^2-n)/2$ if we set the average of the pairwise distance measurements between vehicle i and vehicle j.

The extended Kalman filter algorithm can be viewed as the following set of recursive relationships:

$$\mathbf{A}_{k|k-1} = \mathbf{A}_{k-1|k-1} + T_s \mathbf{u}_{k-1} \tag{9}$$

$$\mathbf{P}_{k|k-1} = \mathbf{Q}_{k-1} + T_s^2 \mathbf{\Gamma}_{k-1} + \mathbf{P}_{k-1|k-1} \tag{10}$$

$$\mathbf{A}_{k|k} = \mathbf{A}_{k|k-1} + \mathbf{K}_k \left(\mathbf{z}_k - \mathbf{h}_k(\mathbf{A}_{k|k-1})\right) \tag{11}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \hat{\mathbf{H}}_k \mathbf{P}_{k|k-1} \tag{12}$$

where

$$p(\boldsymbol{A}_{k-1}|\mathbf{z}_{1:k-1}) \approx \mathcal{N}(\boldsymbol{A}_{k-1}; \mathbf{A}_{k-1|k-1}, \mathbf{P}_{k-1|k-1}) \tag{13}$$

$$p(\boldsymbol{A}_k|\mathbf{z}_{1:k-1}) \approx \mathcal{N}(\boldsymbol{A}_k; \mathbf{A}_{k|k-1}, \mathbf{P}_{k|k-1}) \tag{14}$$

$$p(\boldsymbol{A}_k|\mathbf{z}_{1:k}) \approx \mathcal{N}(\boldsymbol{A}_k; \mathbf{A}_{k|k}, \mathbf{P}_{k|k}) \tag{15}$$

and $\mathcal{N}(x;\mu,P)$ is a Gaussian density with argument x, mean μ and covariance P; $\mathbf{\Gamma}_{k-1}$ the covariance matrix describing the uncertainty in the velocity measurements; $\mathbf{K}_k$ is the Kalman filter gain, defined as:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\hat{\mathbf{H}}_k^T \left(\hat{\mathbf{H}}_k \mathbf{P}_{k|k-1} \hat{\mathbf{H}}_k^T + \mathbf{R}_k\right)^{-1} \tag{16}$$

Given the above set of equations for the Kalman filter, our algorithm can be summarized as follows:

Step 1: Each vehicle performs inter-vehicle distance measurements, and takes a reading of its own velocity. This information is then shared with all vehicles within the cluster. If new vehicles have just joined the cluster, an initial estimate of there position is also required. This initial estimate can be established in one of two ways, either by vehicles exchanging GPS position estimates or measuring and exchanging the inter-vehicle distances to perform trilateration. In general, GPS is the preferred method of establishing estimates, since with trilateration using noisy inter-vehicle distance measurements can often yield ambiguous position estimates [6].

Step 2: Update the prediction equations, (9) and (10). Note, the rank of the state matrix will be dynamic, since the number of vehicles within the cluster can change from one time step to the next.

Step 3: Incorporate the measurements from step 1 to update (11) and (12). The current position estimate based on vehicle movements and a current set of inter-vehicle distance estimates will be contained in the matrix of positions estimates, $\mathbf{A}_{k|k}$, with an associated level of uncertainty captured by the $\mathbf{P}_{k|k}$ matrix. The algorithm also forces the position estimates to be within the road boundaries.

Step 4: Repeat steps 1-3, at the update rate of the filter, T_s.

V. Analysis

A. Algorithm Performance Bounds

To gain a better understanding of the performance bottlenecks of our algorithm, we derive the Cramér-Rao Bound (CRB). The CRB is a classical result from estimation theory that provides a lower bound on the error covariance matrix of any unbiased estimate of unknown parameters. It is a tight bound in the sense that the maximum likelihood detector asymptotically approaches the performance CRB for high signal-to-noise ratio [13]. For the case of localization where velocity information is available and using the motion model (2), the lower estimation bound denoted as $\boldsymbol{P}_k$ is calculated by the recursion [14]:

$$\mathrm{Cov}(\boldsymbol{A}_k) \geq \boldsymbol{P}_k \tag{17}$$

$$\boldsymbol{P}_{k+1} = \left((\boldsymbol{P}_k + T_s\mathbf{Q}_k)^{-1} + \mathbf{J_P}\right)^{-1} \tag{18}$$

where $\mathbf{J_P}$ is equal to the Fisher Information Matrix for the static localization case, studied for the received signal strength, the time-of-arrival, and the angle-of-arrival ranging techniques in [12]. The recursive equation (18) has the form of a Riccati recursive equation and will have a stationary value of [14]:

$$\bar{\boldsymbol{P}} = \left((\bar{\boldsymbol{P}} + T_s\mathbf{Q})^{-1} + \mathbf{J_P}\right)^{-1} \tag{19}$$

which has the following solution:

$$\bar{\boldsymbol{P}} = -\frac{1}{2}T_s\mathbf{Q} + \mathbf{J_P}^{-\frac{1}{2}} \times \left(\mathbf{J_P^{\frac{1}{2}}}(T_s\mathbf{Q} + \frac{T_s^2}{4}\mathbf{Q}\mathbf{J_P}\mathbf{Q})\mathbf{J_P^{\frac{1}{2}}}\right)^{\frac{1}{2}} \mathbf{J_P}^{-\frac{1}{2}} \tag{20}$$

Note, how if $\mathbf{Q} \rightarrow \infty$ — that is as the uncertainty in the mobility goes to infinity — then (20) gives $\bar{\boldsymbol{P}} = \mathbf{J_P}$. Therefore, it is clear how including mobility in the localization is important for improving position estimates. In general, for a linear system with Gaussian noise distribution, the Kalman filter can be shown to achieve this CRB [13]. In our case, we are deploying an extended Kalman filter (EKF) where we are linearizing the nonlinear measurement equation using the Taylor series. Although, generally speaking the extended Kalman filter will not achieve the CRB, it still serves as a good ultimate bound. Generally, how close the EFK comes to the bound is based on how well the Taylor series expansion describes the nonlinear measurement equation.

To see how well our algorithm performs in relation to the bound, we will first consider when there is no hard constraints imposed to force position estimates to be within the confines of the road. Therefore, we can use (20) to compare to the output of our algorithm. Consider, the case where we use a time of arrival based radio ranging scheme with a standard deviation of 4m of error in the inter-vehicle distance estimates. From Figure 1, it is clear that due to the Taylor series approximation, our algorithm without hard lane constraints (the top curve) does not always achieve the CRB (the middle curve). However, when hard lane constraints are imposed the bound (20) can be exceeded (as seen by the lower curve). This result makes sense since by imposing hard lane constraints there is a new piece of information, therefore allowing our approach to exceed the CRB of the general velocity case. Indeed, the true CRB should be re-derived for the case with lane constraints.

B. Computational Cost

As with any localization scheme, it is important to have a high computational efficiency. Let n be the number of vehicles within the local cluster. Our algorithm grows linearly as new vehicles join the cluster. The computational complexity of our algorithm is $\mathcal{O}(m^{2.4} + n^2)$ [15], where m is the size of our measurement vector and n is the number of vehicles within the cluster. The $m^{2.4}$ is due to the multiplication of the $\hat{\mathbf{H}}_k$ matrix and the n^2 is for other matrix multiplications in the Kalman update. If all measurements are used, then m will equal $(n^2 - n)$, however in practice the distance measured between vehicle i and vehicle j are averaged so that m is effectively equal to $(n^2 - n)/2$. However, for the majority of practical scenarios m will be small. As a point of reference, our algorithm implemented in MATLAB runs in near real-time on 3GHz machine, when there are up to 15 vehicles in a cluster.

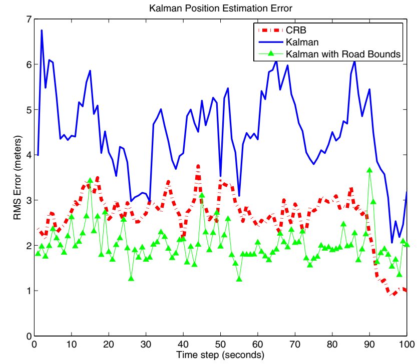

Fig. 1. Cramér-Rao bound for position estimates versus our Kalman filter based solution, lower curve shows performance gain by forcing the position estimate to be within the confines of the road

The only stage in the algorithm that entails communication overhead is the step where the vehicles share distance and velocity estimates with their neighbours. If we assume that non-overlapping clusters do no share the same channel because of range limitations then the communication overhead is $\mathcal{O}(n^2)$, since n^2 measurements are being shared. In practice, this is implemented by each vehicle sending one packet of size $\mathcal{O}(n)$ containing all the observed distance measurements and a packet of constant size for the vehicle's velocity information

VI. Experimentation Results

We study the performance of our algorithm under three scenarios. First, we compare our algorithm to *robust quads* proposed by Moore et al. [6]. The *robust quads* algorithm is designed primarily for localizing a stationary set of nodes, therefore does not have a mechanism for incorporating past localization results. To compare the *robust quads* algorithm to ours, we operate it in a sequential fashion such that at each time step the localization algorithm is run independent of the previous localization results. Next, we will compare our algorithm to the one presented in [8], where velocity and road boundary information is used to localize vehicles with a nonlinear least squares optimization. Therefore, unlike the robust quads implementation, past vehicle locations are incorporated into the estimation of the position at the current time step.

In the next two sections, we will outline the simulation environment used, as well as define the performance metrics

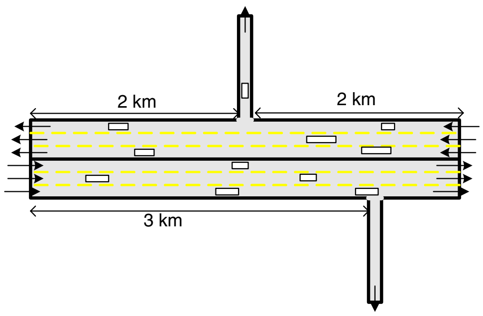

Fig. 2. Roadway for simulated vehicular environment

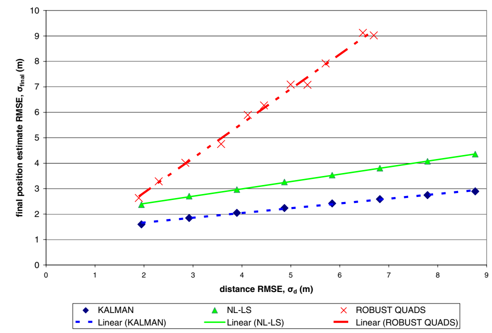

Fig. 3. Showing effect on performance when the inter-vehicle distance estimation error is varied

that we will be using to analyze and compare our localization algorithm to previously proposed ones.

A. Simulation Environment

We have used a microscopic transportation simulator CORSIM (CORridor SIMulator), which has been developed by the US Federal Highway Administration to model vehicle movements. We have modeled a 4 km road with 3 east bound and 3 west bound lanes with vehicles entering into the east and west end of system at the rate of 1200 vehicles per hour depicted in Figure 2. The speed limit for the road was set to 100 km/h.

For the experiments, we track a single vehicle's ability to determine the relative position of all vehicles within its cluster (i.e. all vehicle's within its communication range of 150 m) as it traverses the 4 km section of road. Naturally, there will a variable number of vehicles within a vehicle's communication range as it transverses the network, given this experiment setup, we found that each vehicle generally had between 8 and 14 neighbouring vehicles at each time step.

Also, note that for the simulation experiments we have assumed that the position estimate for a vehicle when it first enters the cluster has been made via GPS, for the algorithms requiring an initial position estimate of the vehicle. After a vehicle has appeared once in the cluster, all future position estimates are based on prior estimates. The GPS position estimate for these experiments was set to differ from the true position by a Gaussian distributed random variable with standard deviation of 6 meters, for consistency with real GPS error levels of 3 to 10 meters [3].

B. Performance Metrics

There are two main metrics we have chosen to use for evaluating the effectiveness of the localization algorithms. The first metric we have used is the root-mean-square error (RMSE) in the final position estimate, which can be thought of as the average distance of the final position estimate from the actual position. We have defined RMSE in the final position estimate as follows:

$$\sigma_{\text{final}} = \sqrt{\sum_{i=1}^{n} \frac{(x_{\text{final est. i}} - x_{\text{actual i}})^2 + (y_{\text{final est. i}} - y_{\text{actual i}})^2}{n}} \tag{21}$$

where n is the number of vehicles in the cluster at the current time; $(x_{\text{actual i}}, y_{\text{actual i}})$ represents the actual position of vehicle i (often referred to as the ground truth in other literature); $(x_{\text{final est. i}}, y_{\text{final est. i}})$ is the position estimate of vehicle i after running step 4 of our algorithm. Note, that since only one vehicle is fixed at position $(0, 0)$ when the position estimate is being made, an arbitrary reflection or rotation from the actual positions of the vehicles is an equally valid solution.

The second metric we have used is the RMSE in the inter-vehicle distance measurements, which we have defined as:

$$\sigma_{\text{d}} = \sqrt{\sum_{i,j=1}^{M} \frac{(\hat{d_{i,j}} - d_{i,j})^2}{M}} \tag{22}$$

where M is the number of inter-vehicle distance measurements, $\hat{d_{i,j}}$ is the measured distance between vehicles i and j, and $d_{i,j}$ is the actual distance between vehicles i and j. This metric is important because it allows us to measure the sensitivity of the localization algorithm to error levels in the distance measurements.

C. Performance Comparisons with Other Algorithms

First we will compare the performance of our algorithm against the robust quads algorithm presented in [6]. Since, this algorithm does not rely on initial position estimates of the vehicle, we will compare it to ours based on changing the error levels in the inter-vehicle distance measurements.

To examine the effect of errors in the inter-vehicle distance estimates on the final position estimate, we have varied the RMSE in the distance measurements while leaving all other parameters constant. The results of this experiment are presented in Figure 3, where each data point is the average of 10 runs of the algorithms through our simulated vehicular environments (as was described in the section entitled *Simulation Environment*). The top curve shows the robust quads algorithm and the bottom curve is our proposed algorithm. From the figure, it is clear the error level in position estimates of the robust quads algorithm grows at a linear rate which is on the order of 7 times greater than our algorithm. The major difference in performance can be attributed to the fact that the robust quads algorithm does not make use of velocity information or road constraints.

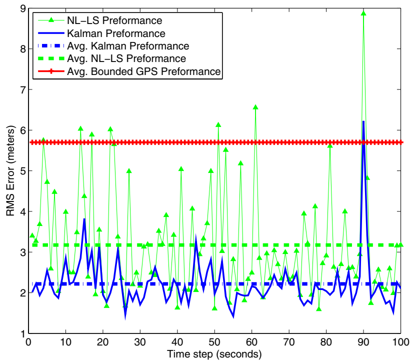

Fig. 4. Comparison of Nonlinear Least of Squares approach to our Kalman filter based approach, also the performance of using GPS with a mapping module is shown for reference

To verify this claim and to see the benefits of road and velocity information we will compare our algorithm to the non-linear least squares approach presented in [8].

Comparing the middle curve in Figure 3, which is the results of the algorithm presented in [8], to the upper curve showing the robust quads algorithm [6], it is clear that by including velocity and road constraints the sensitivity to distance measurement errors is reduced. Although, the algorithm presented in [8] shows significant performance gains over existing static based localization schemes by including these additional constraints, on average it performs worse than the algorithm proposed here. The primary reason for this performance differential is in how the velocity and road constraints are incorporated into the optimization scheme, as discussed in section II of this paper. This can be verified by examining the result of how well a single vehicle was able to localize its neighbours at each time step as it traveled through the system. If we assume that a Gaussian random variable with standard deviation of 6 meters describes the noise characteristics of the inter-vehicle distance measurements, then we get the results shown in Figure 4. The curve with the higher peaks is the nonlinear least squares implementation proposed in [8] and the bottom curve is our Kalman filter based approach presented in the this paper. Overall, we see that there is much less variation in the error levels of our algorithm versus the non-linear least squares (NL-LS) approach. This lower variation results in an overall lower average level of error for our algorithm — 2.2m for the Kalman filter approach versus 3.2m for the NL-LS approach.

Also, in Figure 4, for reference we have included the average performance of a GPS based system (i.e. the horizontal line with RMS error of approximately 5.7m), which made use of lane and velocity constraints to establish position estimates. The GPS system was set to have an average accuracy of 6 meters, which is consistent with the performance of real-world GPS [3], then the velocity lane constraints were imposed. This reinforces the additional benefits of using inter-vehicle distance estimates.

VII. CONCLUSION

We have shown that gains in accuracy and reliability can be achieved over GPS-based approaches by using inter-vehicle distance measurements taken by a radio based ranging technology. Also that the accuracy of previously proposed radio ranging based localization can be improved by taking into account extra information that is available to vehicles (e.g. maps of the road; vehicle kinematics). We have performed an analysis of our algorithm to examine its ultimate accuracy performance limits, computational complexity, and its sensitivity measurement errors. We have shown that our Kalman filter based solution is accurate and reliable, and allows a real-time implementation. The structure of our algorithm allows us to place a probabilistic confidence on the position estimates, so that inaccurate position estimates can be actively identified making it very practical for future vehicle safety applications.

REFERENCES

[1] W. Xiang, P. Richardson, and J. Guo, "Introduction and preliminary experimental results of wireless access for vehicular environments (WAVE) systems," in *IEEE V2VCOM*, July 2006.
[2] R. Tatchikou, S. Biswas, and F. Dion, "Cooperative vehicle collision avoidance using inter-vehicle packet forwarding," in *Proceedings of Global Telecommunications Conference, 2005. GLOBECOM '05.* IEEE, November 2005.
[3] C. Drane and C. Rizos, "Positionng systems in intelligent transportation systems," *Artech House, Boston London*, pp. 230–231, 1997.
[4] J. Hightower and G. Borriella, "A survey and taxonomy of location systems for ubiquitous computing," *IEEE Computer*, pp. 57–66, August 2001.
[5] S. Capkun, M. Hamdi, and J. Hubaux, "GPS-free positioning in mobile ad-hoc networks," in *Proceedings of Hawaii Int. Conf. on System Sciences*, January 2001.
[6] D. Moore, J. Leonard, D. Rus, , and S. Teller, "Robust distributed network localization with noisy range measurement," in *Proceedings of The Second ACM Conference on Embedded Networked Sensor Systems (SenSys)*, November 2004.
[7] W. Ruml and Y. Shang, "Improved MDS-based localization," *INFOCOM 2004. Twenty-third AnnualJoint Conference of the "IEEE" Computer and Communications Societies*, vol. 4.
[8] R. Parker and S. Valaee, "Vehicle localization in vehicular networks," *IEEE Vehicular Technology Conference*, September 2006.
[9] ——, "Robust min-max localization algorithm," *IEEE Intelligent Transportation Systems Conference*, September 2006.
[10] L. Hu and D. Evans, "Localization for mobile sensor networks," *ACM MobiCom '04*, October 2004.
[11] A. Savvides, M. Srivastava, S. Basagni, M. Conti, S. Giordano, and I. S. (Eds.), "Mobile ad hoc networking," *IEEE Press Wiley, New York*, pp. 233–234, December 2003.
[12] N. Patwari, J. N. Ash, A. O. Hero, R. L. Moses, and N. S. Correal, "Cooperative localization in wireless sensor networks," *IEEE Signal Processing Maganize*, July 2005.
[13] H. L. V. Trees, *Detection, Estimation, and Modulation Theory: Part I.* Wiley, New York, 1968.
[14] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks," *IEEE Signal Processing Magazine*, July 2005.
[15] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.