A New Approach to Mobile Position Tracking[1]

Sinan Gezici, Hisashi Kobayashi and H. Vincent Poor
Department of Electrical Engineering
Princeton University
Princeton, NJ 08544
{sgezici,hisashi,poor}@princeton.edu

***Abstract*— Recently, there has been much interest in accurate determination of mobile user locations in cellular environments. A general approach to this geolocation problem is to gather measurements from a number of base stations and to estimate user locations using a least squares approach [1]. However, in non-line-of-sight (NLOS) situations, measurements are significantly biased. Hence, very large errors in location estimation may be introduced. In this paper, an approach developed in learning theory, namely, Support Vector Regression (SVR), is applied to the geolocation problem with the addition of Kalman-Bucy filtering to smooth location estimates in a mobile tracking scenario.**

I. Introduction

Recently, the subject of mobile positioning in wireless communication systems has drawn considerable attention. With accurate location estimation, a variety of new applications and services such as Enhanced-911, location sensitive billing, improved fraud detection, intelligent transport system (ITS) and improved traffic management will become feasible [2].

Mobile positioning using radiolocation techniques usually involves time of arrival (TOA), time difference of arrival (TDOA), angle of arrival (AOA), signal strength (SS) measurements or some combination of these methods. Multipath, non-line-of-sight (NLOS) propagation and multiple access interference are often the main sources of errors in geolocation, and make mobile positioning challenging. Among these error sources, NLOS is probably the most crucial one. Several geolocation techniques have been proposed in the literature which can reduce the effect of NLOS error to some extent [3]-[5]. However, very precise location estimation is usually not possible with these methods. In order to achieve smaller errors, formulating the geolocation problem as a learning problem seems to be a promising approach.

In this paper, a two-step location estimation algorithm, shown in Figure 1, is proposed to estimate the location of a mobile user. The first step employs an approach developed in learning theory, namely, "support vector regression" (SVR), which is used to obtain an initial estimate of the mobile location. In this step, measurements are taken at known locations *a priori* and a training set database is created. Then, a measurement of the mobile of interest is taken and the SVR technique is applied to this training set to generate an estimate of the mobile's location. Unlike conventional techniques which are based on the "empirical risk minimization" principle, the SVR technique uses the so-called "structural risk minimization" principle. That is, it minimizes an upper bound on the expected risk instead of minimizing the empirical risk directly. Thus, this technique is superior to other conventional learning algorithms, and application of the SVR technique has yielded very good results in regression estimation problems (e.g. [10]).

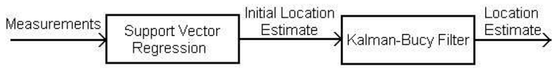

Fig. 1. The location estimation algorithm.

Moreover, the use of a training set provides very accurate location estimation and the presence of NLOS errors does not affect the final result significantly. It is important, however, to note that the main underlying assumption of this method is that the training set remains relevant during the evaluation period. In other words, the environment is assumed to be sufficiently stable so that the measurements taken for the database creation and those for the mobile location estimation are associated with almost the same error statistics.

In the second step of our algorithm, a Kalman-Bucy filter is employed, which takes the location estimates obtained from the SVR method as inputs. The Kalman-Bucy filter allows accurate tracking of the mobile and reduces its error by smoothing the location estimates.

The remainder of this paper is organized as follows. Section 2 describes the SVR method that is used for the initial estimation of mobile location. The application of the Kalman-Bucy filter to smooth the results is explained in Section 3, followed by simulation results in Section 4. Finally, Section 5 presents some concluding remarks.

II. Support Vector Regression

Consider a training set consisting of input-output pairs

$$A = \{(\mathbf{m}_1, p_1), (\mathbf{m}_2, p_2), ...(\mathbf{m}_k, p_k)\} \quad (1)$$

where $\mathbf{m} \in \mathbb{R}^n$ and $p \in \mathbb{R}$.

In order to do non-linear regression, data $\mathbf{m}$ is mapped into a higher dimensional feature space $\mathcal{F}$ and linear regression is performed in this space [6]; i.e.,

$$p = f(\mathbf{m}) = \phi(\mathbf{m})^T \mathbf{w} + b, \quad \phi : \mathbb{R}^n \rightarrow \mathcal{F}, \, \mathbf{w} \in \mathcal{F}. \quad (2)$$

[1]This research is supported in part by the U.S. National Science Foundation under grant CCR-99-79361, and in part by the New Jersey Center for Wireless Telecommunications.

where ϕ is a suitably defined non-linear mapping, b is a bias term and $\mathbf{w}$ is a coefficient vector[2].

The SVR technique estimates the coefficient vector, $\mathbf{w}$, by minimizing the following structural risk:

$$R(\mathbf{w}) = \sum_{i=1}^{k} g(f(\mathbf{m}_i) - p_i) + \frac{1}{2C}\|\mathbf{w}\|^2, \tag{3}$$

where the first term is the empirical risk with $g(.)$ denoting a cost function, and the second term quantifies complexity of the feature space. Therefore, the constant C controls the trade-off between the complexity of the regressor and the empirical error. When C is chosen to be very large, the main aim becomes minimizing the empirical error, which means fitting the training data as closely as possible. However, when the training set does not contain many elements, a smaller value of C should be chosen so that the data is approximated with a less complex regressor that can fit to new data better, which means that the regressor generalizes better.

We consider the ϵ-insensitive loss function as the cost function in (3), i.e.,

$$g(f(\mathbf{m}) - p) = \begin{cases} |f(\mathbf{m}) - p\,| - \epsilon & \text{if } |f(\mathbf{m}) - p\,| \geq \epsilon \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Then, the optimization problem (3) can be expressed, after use of some slack variables and the Lagrange multiplier technique, as follows [8]:

$$\max_{\alpha,\, \alpha^*} \quad \sum_{i=1}^{k} [\alpha_i^*(p_i - \epsilon) - \alpha_i(p_i + \epsilon)]$$
$$-\frac{1}{2}\sum_{i=1}^{k}\sum_{j=1}^{k}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(\mathbf{m}_i, \mathbf{m}_j)$$

subject to constraints

$$\sum_{i=1}^{k}(\alpha_i - \alpha_i^*) = 0\,; \quad 0 \leq \alpha_i,\, \alpha_i^* \leq C \;\; i = 1, 2, ..., k \tag{5}$$

and

$$\mathbf{w} = \sum_{i=1}^{k}(\alpha_i - \alpha_i^*)\phi(\mathbf{m}_i), \tag{6}$$

where α_i^* and α_i are Lagrange multipliers and $K(\mathbf{m}_i, \mathbf{m}) = \phi(\mathbf{m}_i)^T\phi(\mathbf{m})$ is the kernel function.

Using (6), equation (2) can be expressed as

$$p = f(\mathbf{m}) = \sum_{i=1}^{k}(\alpha_i - \alpha_i^*)K(\mathbf{m}_i, \mathbf{m}) + b, \tag{7}$$

where the radial basis function (RBF) given by

$$K(\mathbf{m}_i, \mathbf{m}) = \exp(-\|\mathbf{m}_i - \mathbf{m}\|^2/(2\sigma^2)) \tag{8}$$

is used as the kernel function.

The bias term b can be calculated using the points $\mathbf{m}_k$ on the ϵ-margin since the prediction error for those points is known to be $\delta_k = \epsilon\,\mathrm{sign}(\alpha_k - \alpha_k^*)$. It can be shown using the Lagrangian for (5) and Karush-Kuhn-Tucker (KKT) conditions that the points $\mathbf{m}_k$ on the margin correspond to α_k and α_k^* in the open interval $(0, C)$ [9]. Then, b can be simply calculated as the average value of $[\delta_k + p_k - \sum_i(\alpha_k - \alpha_k^*)K(\mathbf{m_i}, \mathbf{m_k})]$ for $\alpha_k, \alpha_k^* \in (0, C)$ [7].

After solving the optimization problem (5) and calculating b as described above, the output p corresponding to a new input vector $\mathbf{m}$ can be estimated using (7). For the geolocation problem, the input vector $\mathbf{m}$ denotes the measurement vector, which can, for example, consist of TOA, TDOA, AOA or SS measurements, and the output value p denotes one coordinate of the mobile location.

III. Tracking the Mobile

When the aim is to track a specific mobile station, some smoothing operation on the location estimates, obtained as described in the previous section, is needed. This smoothing operation is achieved by a Kalman-Bucy filter.

Assume that each base station (BS) takes measurements from the mobile every $\triangle t$ seconds. The location estimate obtained from the SVR method at time t is denoted by $\mathbf{Y}(t)$ where

$$\mathbf{Y}(t) = [Y_1(t) \quad Y_2(t)]^T. \tag{9}$$

Let the state vector be defined as

$$\mathbf{X}(t) = [X_1(t) \quad X_2(t) \quad X_3(t) \quad X_4(t)]^T \tag{10}$$

where $X_1(t)$ and $X_2(t)$ denote the x and y coordinates, respectively, of the mobile location, whereas $X_3(t)$ and $X_4(t)$ denote the x and y coordinates, respectively, of the velocity vector at time t. Then, the state and measurement equations can be expressed as follows:

$$\mathbf{X}_{i+1} = \mathbf{F}_i\mathbf{X}_i + \mathbf{G}_i\mathbf{U}_i \tag{11}$$

$$\mathbf{Y}_i = \mathbf{H}_i\mathbf{X}_i + \mathbf{V}_i \tag{12}$$

for $i = 0, 1, ...$ where i is the time at which the i^{th} location estimation is performed, $\mathbf{F}_i$, $\mathbf{G}_i$ and $\mathbf{H}_i$ are the following matrices:

$$\mathbf{F}_i = \begin{bmatrix} 1 & 0 & \triangle t & 0 \\ 0 & 1 & 0 & \triangle t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{G}_i = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \triangle t & 0 \\ 0 & \triangle t \end{bmatrix} \mathbf{H}_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}^T$$

$\mathbf{U}_i$ is the two dimensional random acceleration component which is modelled by a zero mean Gaussian process, and $\mathbf{V}_i$ is the two-dimensional zero mean Gaussian measurement error.

For the linear stochastic system of (11) and (12), the discrete-time Kalman-Bucy filter equations can be used assuming that $\mathbf{U}_i$ and $\mathbf{V}_i$ are independent sequences that are also independent from the initial state. The estimated state at time i given i measurements is the conditional expectation of the state given those previous measurements $(\mathbf{Y_0}, \mathbf{Y_1}, ..., \mathbf{Y_i})$, which is simply denoted as follows:

$$\hat{\mathbf{X}}_{i|i} = E\{\mathbf{X}_i|\mathbf{Y}_0^i\}. \tag{13}$$

[2] $\mathbf{x}^T\mathbf{y}$ denotes the inner product of two vectors

The estimates $\hat{\mathbf{X}}_{i|i} = E\{\mathbf{X}_i|\mathbf{Y}_0^i\}$ and $\hat{\mathbf{X}}_{i+1|i} = E\{\mathbf{X}_{i+1}|\mathbf{Y}_0^i\}$ are given recursively by the following equations [11]:

$$\hat{\mathbf{X}}_{i|i} = \hat{\mathbf{X}}_{i|i-1} + \mathbf{K}_i(\mathbf{Y}_i - \mathbf{H}_i\hat{\mathbf{X}}_{i|i-1}) \quad i = 0, 1, ..., \tag{14}$$

and

$$\hat{\mathbf{X}}_{i+1|i} = \mathbf{F}_i\hat{\mathbf{X}}_{i|i} \quad i = 0, 1, ..., \tag{15}$$

where the Kalman gain matrix $\mathbf{K}_i$ is given by

$$\mathbf{K}_i = \mathbf{\Sigma}_{i|i-1}\mathbf{H}_i^T(\mathbf{H}_i\mathbf{\Sigma}_{i|i-1}\mathbf{H}_i^T + \mathbf{R}_i)^{-1}, \tag{16}$$

with $\mathbf{\Sigma}_{i|i-1} = Cov(\mathbf{X}_i|\mathbf{Y}_0^{i-1})$ and $\mathbf{R}_i = Cov(\mathbf{V}_i)$. Here, $\mathbf{\Sigma}_{i|i-1}$ represents the covariance matrix of the prediction error $\mathbf{X}_i - \hat{\mathbf{X}}_{i|i-1}$, conditioned on $\mathbf{Y}_0^{i-1}$. This matrix and the filtering error covariance, $\mathbf{\Sigma}_{i|i} = Cov(\mathbf{X}_i|\mathbf{Y}_0^i)$, can be computed using the following recursion:

$$\mathbf{\Sigma}_{i|i} = \mathbf{\Sigma}_{i|i-1} - \mathbf{K}_i\mathbf{H}_i\mathbf{\Sigma}_{i|i-1} \quad i = 0, 1, ..., \tag{17}$$

$$\mathbf{\Sigma}_{i+1|i} = \mathbf{F}_i\mathbf{\Sigma}_{i|i}\mathbf{F}_i^T + \mathbf{G}_i\mathbf{Q}_i\mathbf{G}_i^T \quad i = 0, 1, ..., \tag{18}$$

where $\mathbf{Q}_i = Cov(\mathbf{U}_i)$.

With proper initial values, equations (14) through (18) can be used to evaluate the state of the mobile at each time. The initial value for the state variable estimate can be set as follows:

$$\hat{\mathbf{X}}_{0|0} = [\hat{x}_1(i) \quad \hat{x}_2(i) \quad 0 \quad 0]^T, \tag{19}$$

where $\hat{\mathbf{x}} = [\hat{x}_1(i) \quad \hat{x}_2(i)]^T$ is the first location estimate obtained by the SVR method, and the initial velocity is assumed to be zero.

IV. Simulation Results

Geolocation accuracy for a mobile moving in a microcell area will be simulated in this section. In the simulations, SVR is implemented by using MATLAB Support Vector Machine Toolbox [12].

A 600 $\times$ 600 m relevant area with three base stations at locations (187.5, 150), (487.5, 150) and (337.5, 409.8) are considered. The mobile is moving from location (155, 305) to (575, 305) on the solid track shown in Figure 2 with a constant speed of 20 m/s. TOA measurements are taken at a rate of 20 samples/s.

The elliptical scattering model is used to generate TOA data. In this model, uniformly distributed scatterers inside an ellipse with foci at the BS and the mobile are assumed. The TOA probability density function is given by [13]:

$$p(x) = \begin{cases} \frac{c(2c^2x^2 - D^2)}{4a_m b_m\sqrt{c^2x^2-D^2}}, & \frac{D}{c} \le x \le \tau_m \\ 0, & \text{else,} \end{cases} \tag{20}$$

where c is the speed of light, D is the true distance between the BS and the mobile and τ_m is the maximum delay in that region; that is, only multipath components that arrive within τ_m seconds are considered. So the semi-major axis parameters of the ellipse, a_m and b_m, can be expressed as

$$a_m = \frac{c\tau_m}{2} \text{ and } b_m = \frac{1}{2}\sqrt{c^2\tau_m^2 - D^2}. \tag{21}$$

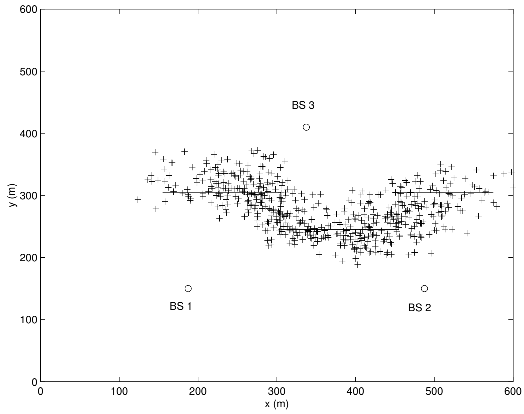

Fig. 2. The simulation environment, the track of the mobile and the SVR estimates.

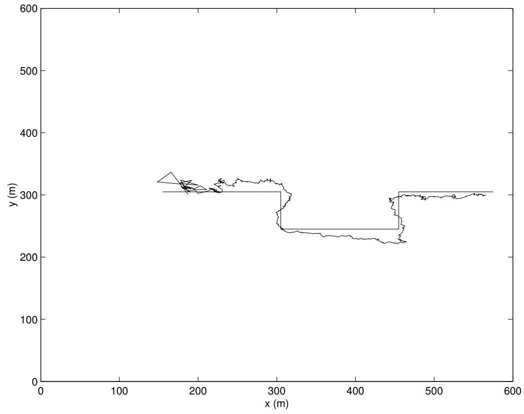

Fig. 3. The location estimates after Kalman-Bucy filtering.

This TOA distribution is used in the simulation by equating τ_m to a multiple of the true TOA between the location of interest and the BS, that is $\tau_m = \alpha D/c$, so that closer locations have higher probability of having smaller NLOS errors, which is a reasonable assumption.

For the SVR technique, the RBF with $\sigma = 400$ is used as the kernel and the ϵ-insensitive cost function with $\epsilon = 1$ is employed.

In the first set of simulations, the mobile is tracked in the solid line shown in Figure 2 with $C = 275$ in (3), $\alpha = 1.5$ and $\Delta = 50$ m, where Δ is the distance between the consecutive training locations. The location estimates obtained by the SVR method are also shown in Figure 2 and the estimates from the Kalman-Bucy filter, which takes the SVR estimates as inputs, are presented in Figure 3. From these two figures, the effect of Kalman-Bucy filtering is seen clearly. The average error[3] between the true and the estimated locations is 37.8 m for the SVR estimates, which improves to 21.1 m after Kalman-Bucy filtering. The velocity estimates become close to 20 m/s after just a few seconds as shown in Figure 4.

[3]The average error is calculated as the average of the Euclidean distances between the true locations and the location estimates.

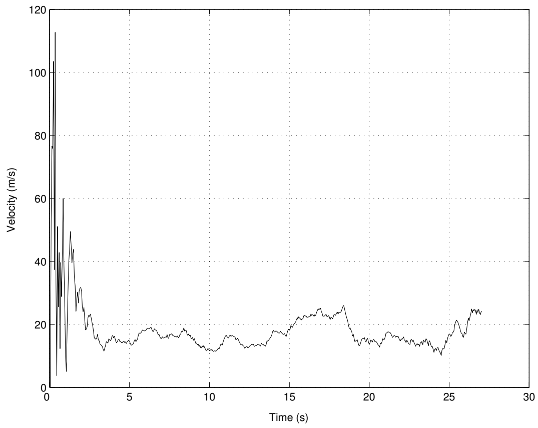

Fig. 4. The velocity estimates for the mobile.

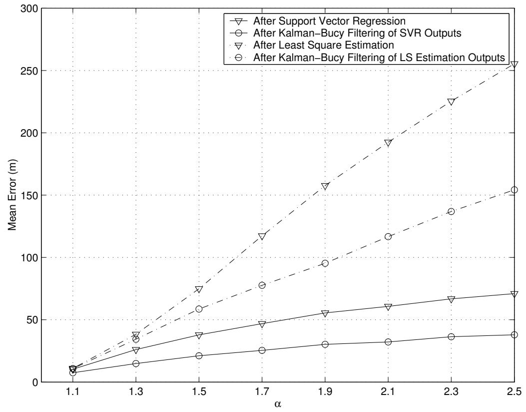

Fig. 5. Average error versus α.

In Figure 5, the average error (obtained by averaging 50 independent runs) is plotted for different values of α with $\Delta =$ 50m and $C = 275$. Also the estimates from the conventional least squares algorithm are plotted. As expected, the average error increases with α. However, it is also seen that even in very harsh NLOS conditions, in which the NLOS error is 1.5 times the true distance (that is $\alpha = 2.5$), the average error is less than 40 m for our technique in contrast with the large error of 150 m of the least square algorithm.

The last simulation compares the error values for different distances between the training samples; that is, for different Δ values with $\alpha = 1.5$. In Table 1, k denotes the number of training data pairs, C denotes the trade-off constant in (3), Error-1 denotes the average error after the SVR technique and Error-2 denotes the average error after Kalman-Bucy filtering. Note that for smaller training data sizes smaller C values are chosen so that the regressor does not fit the data very much and generalizes better. It is seen from the table that the error increases with Δ since higher Δ values mean fewer training samples (averages over 50 independent runs are plotted) and the error of our technique with a training size as low as 9 has almost the same final error as the LS technique.

Δ (m)	k	C	Error-1 (m)	Error-2 (m)
50	169	275	37.8	21.1
100	49	250	41.1	25.4
200	16	225	47.5	37.4
300	9	225	63.1	58.7

TABLE I

AVERAGE ERRORS FOR DIFFERENT TRAINING SET SIZES

V. CONCLUSION

In this paper, SVR and Kalman-Bucy filtering are used to estimate the location of a mobile. The main advantage of the method is that it yields very accurate estimates even in NLOS environments and the use of the SVR technique provides good results even with training data set of relatively small size. The main assumption in this model is that the environment is stationary enough to have almost the same noise statistics throughout the experiments. When the environment is not sufficiently stable, the database (that is, the training set) needs to be updated periodically. Moreover, taking data from all locations may be quite expensive. However, this method may be used in some special areas, where there exist significant NLOS errors and yet accurate location estimation is crucial.

REFERENCES

[1] J. Caffery, Jr. and G. Stuber, "Subscriber location in CDMA cellular networks," *IEEE Trans. Vehicular Technology*, vol. 47, pp. 406-416, May 1998.

[2] J. Caffery, Jr., *Wireless Location in CDMA Cellular Radio Systems,* Kluwer Academic Publishers, Boston, 2000.

[3] M. P. Wylie and J. Holtzman, "The non-line of sight problem in mobile location estimation," *Proc. 5th IEEE International Conference on Universal Personal Communications, 1996*, vol. 2, pp. 827-831.

[4] M. P. Wylie-Green and S. S. Wang, "Robust range estimation in the presence of non-line-of-sight error,"*Proc. IEEE Vehicular Technology Conference, Fall 2001*, vol. 1, pp. 101-105.

[5] P. C. Chen, "A non-line-of-sight error mitigation algorithm in location estimation,"*Proc. IEEE Wireless Communications and Networking Conference, 1999*, vol. 1, pp. 316-320.

[6] V. Vapnik, S.E. Golowich, and A. Smola. Support vector method for function approximation, regression estimation, and signal processing. In M. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processings Systems 9*, MIT Press, Cambridge, MA, 1997, pp. 281-287.

[7] K. Muller, A. J. Smola, G. Ratsch, B. Scholkopf, J. Kohlmorgen, and V. Vapnik, "Predicting Time Series with Support Vector Machines,"*Proc. International Conference on Artificial Neural Networks, 1997*, pp. 999-1004.

[8] V. Vapnik, *The Nature of Statistical Learning Theory,*" Springer-Verlag, New York, 1995.

[9] W. Chu, S. S. Keerthi, and C. J. Ong, "Bayesian support vector regression using a unified loss function," Submitted to *IEEE Transaction on Neural Network*.

[10] S. Mukherjee, E. Osuna, and F. Girosi, "Nonlinear prediction of chaotic time series using support vector machines," *Proc. IEEE Neural Networks for Signal Processing Workshop, 1997*, pp. 511 -520.

[11] H. V. Poor, *An Introduction to Signal Detection and Estimation,* Springer-Verlag, New York, 1994.

[12] S. R. Gun, "MATLAB Support Vector Machine Toolbox," University of Southampton, 1998. (http://www.isis.ecs.soton.ac.uk/isystems/kernel).

[13] R. B. Ertel and J. H. Reed, "Angle and time of arrival statistics for circular and elliptical scattering models," *IEEE J. Selected Areas in Communications*, vol. 17, no. 11, pp. 1829-1840, Nov 1999.