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 signifi-
cantly 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 com-
munication 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) mea-
 surements or some combination of these methods. Multipath,
non-line-of-sight (NLOS) propagation and multiple access inter-
ference 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 minimiza-
tion” 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]).

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
\[
A = \{(m_1, p_1), (m_2, p_2), \ldots, (m_k, p_k)\}
\]  
where \(m \in \mathbb{R}^n\) and \(p \in \mathbb{R}\).

In order to do non-linear regression, data \(m\) is mapped into
a higher dimensional feature space \(\mathcal{F}\) and linear regression
is performed in this space [6]; i.e.,
\[
p = f(m) = \phi(m)^T w + b, \quad \phi : \mathbb{R}^n \rightarrow \mathcal{F}, \ w \in \mathcal{F}.
\]
where $\phi$ is a suitably defined non-linear mapping, $b$ is a bias term and $w$ is a coefficient vector.$^2$

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

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

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 $\epsilon$-insensitive loss function as the cost function in (3), i.e.,

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

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

$$ \max_{\alpha_i, \alpha_i^*} \sum_{i=1}^{k} [\alpha_i^*(p_i - \epsilon) - \alpha_i(p_i + \epsilon)] - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(m_i, m_j) $$

subject to constraints

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

and

$$ w = \sum_{k} (\alpha_i - \alpha_i^*)\phi(m_i), \quad (6) $$

where $\alpha_i^*$ and $\alpha_i$ are Lagrange multipliers and $K(m_i, m) = \phi(m_i)^T\phi(m)$ is the kernel function.

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

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

where the radial basis function (RBF) given by

$$ K(m_i, m) = \exp(-||m_i - m||^2/(2\sigma^2)) \quad (8) $$

is used as the kernel function.

The bias term $b$ can be calculated using the points $m_i$ on the $\epsilon$-margin since the prediction error for those points is known to be $\delta_k = \epsilon \text{sign}(\alpha_k - \alpha_k^*)$. It can be shown using the Lagrangian for (5) and Karush-Kuhn-Tucker (KKT) conditions that the points $m_i$ on the margin correspond to $\alpha_k$ and $\alpha_k^*$ in the open interval $(0, C)$ [9]. Then, $b$ can be simply calculated as the average value of $[\delta_k + p_k - \sum_{i=1}^{k} (\alpha_k^* - \alpha_k)K(m_i, m_k)]$ 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 $m$ can be estimated using (7). For the geolocation problem, the input vector $m$ 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 $\Delta t$ seconds. The location estimate obtained from the SVR method at time $t$ is denoted by $Y(t)$ where

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

Let the state vector be defined as

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

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:

$$ X_{i+1} = F_i X_i + G_i U_i \quad (11) $$

$$ Y_i = H_i X_i + V_i \quad (12) $$

for $i = 0, 1, ...$, where $i$ is the time at which the $i^{th}$ location estimation is performed, $F_i$, $G_i$, and $H_i$ are the following matrices:

$$ F_i = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad G_i = \begin{bmatrix} 0 & 0 \\ 0 & \Delta t \\ \Delta t & 0 \end{bmatrix}, \quad H_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}. $$

$U_i$ is the two dimensional random acceleration component which is modelled by a zero mean Gaussian process, and $V_i$ 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 $U_i$ and $V_i$ 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 $(Y_0, Y_1, ..., Y_i)$, which is simply denoted as follows:

$$ \hat{X}_{i|i} = E\{X_i | Y_{0:i} \}. \quad (13) $$
The estimates $\hat{X}_{i|j} = E\{X_i|Y_i^k\}$ and $\hat{X}_{i+1|i} = E\{X_{i+1}|Y_i^k\}$ are given recursively by the following equations [11]:

$$\hat{X}_{i|i} = \hat{X}_{i|i-1} + K_i (Y_i - H_i \hat{X}_{i|i-1}) \quad i = 0, 1, \ldots, \quad (14)$$

and

$$\hat{X}_{i+1|i} = F_i \hat{X}_{i|i} \quad i = 0, 1, \ldots, \quad (15)$$

where the Kalman gain matrix $K_i$ is given by

$$K_i = \Sigma_{i|i-1} H_i^T (H_i \Sigma_{i|i-1} H_i^T + R_i)^{-1}, \quad (16)$$

with $\Sigma_{i|i-1} = Cov(X_i|Y_{i-1})$ and $R_i = Cov(V_i)$. Here, $\Sigma_{i|i-1}$ represents the covariance matrix of the prediction error $X_i - \hat{X}_{i|i-1}$, conditioned on $Y_{i-1}$. This matrix and the filtering error covariance, $\Sigma_{i|i} = Cov(X_i|Y_i^i)$, can be computed using the following recursion:

$$\Sigma_{i|i} = \Sigma_{i|i-1} - K_i H_i \Sigma_{i|i-1} \quad i = 0, 1, \ldots, \quad (17)$$

$$\Sigma_{i+1|i} = F_i \Sigma_{i|i} F_i^T + G_i Q_i G_i^T \quad i = 0, 1, \ldots, \quad (18)$$

where $Q_i = Cov(U_i)$.

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:

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

where $\hat{x} = [\hat{x}_1(i) \quad \hat{x}_2(i)]^T$ 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 (187.5, 150), (487.5, 150) and (337.5, 409.8) are simulated in this section. In the simulations, SVR estimates, which improves to 21.1 m after Kalman-Bucy filter, which takes the SVR estimates as inputs.

This TOA distribution is used in the simulation by equating $\tau_m$ 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 $\epsilon$-insensitive cost function with $\epsilon = 1$ is employed.

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]:

$$p(x) = \begin{cases} \frac{1}{4a_m b_m \sqrt{\pi} c^2 \Delta^2} e^{-D \sqrt{c^2 \Delta^2 - D^2}} & \frac{D}{c} \leq x \leq \tau_m \\ 0 & \text{else} \end{cases} \quad (20)$$

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

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

3The average error is calculated as the average of the Euclidean distances between the true locations and the location estimates.
In Figure 5, the average error (obtained by averaging 50 independent runs) is plotted for different values of $\alpha$ with $\Delta = 50\text{m}$ and $C = 275$. Also the estimates from the conventional least squares algorithm are plotted. As expected, the average error increases with $\alpha$. 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 $\Delta$ 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 $\Delta$ since higher $\Delta$ 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.

### Table 1

<table>
<thead>
<tr>
<th>$\Delta$ (m)</th>
<th>$k$</th>
<th>$C$</th>
<th>Error-1 (m)</th>
<th>Error-2 (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>169</td>
<td>275</td>
<td>37.8</td>
<td>21.1</td>
</tr>
<tr>
<td>100</td>
<td>49</td>
<td>250</td>
<td>41.1</td>
<td>25.4</td>
</tr>
<tr>
<td>200</td>
<td>16</td>
<td>225</td>
<td>47.5</td>
<td>37.4</td>
</tr>
<tr>
<td>300</td>
<td>9</td>
<td>225</td>
<td>63.1</td>
<td>58.7</td>
</tr>
</tbody>
</table>

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


