Bluetooth Aided Mobile Phone Localization: A Nonlinear Neural Circuit Approach

SHUAI LI, Stevens Institute of Technology
YUESHENG LOU, Yiwu Industrial and Commercial College
BO LIU, University of Massachusetts, Amherst

It is meaningful to design a strategy to roughly localize mobile phones without a GPS by exploiting existing conditions and devices especially in environments without GPS availability (e.g., tunnels, subway stations, etc.). The availability of Bluetooth devices for most phones and the existence of a number of GPS equipped phones in a crowd of phone users enable us to design a Bluetooth aided mobile phone localization strategy. With the position of GPS equipped phones as beacons, and with the Bluetooth connection between neighbor phones as proximity constraints, we formulate the problem into an inequality problem defined on the Bluetooth network. A recurrent neural network is developed to solve the problem distributively in real time. The convergence of the neural network and the solution feasibility to the defined problem are both theoretically proven. The hardware implementation architecture of the proposed neural network is also given in this article. As applications, rough localizations of drivers in a tunnel and localization of customers in a supermarket are explored and simulated. Simulations demonstrate the effectiveness of the proposed method.

Categories and Subject Descriptors: C.1.3 [**Processor Architectures**]: Other Architecture Styles—*Neural nets*

General Terms: Design, Algorithms, Performance

Additional Key Words and Phrases: Recurrent neural network, Bluetooth, mobile phone, localization, neural circuit

ACM Reference Format:
Shuai Li, Yuesheng Lou, and Bo Liu. 2014. Bluetooth aided mobile phone localization: A nonlinear neural circuit approach. ACM Trans. Embedd. Comput. Syst. 13, 4, Article 78 (February 2014), 15 pages.
DOI: http://dx.doi.org/10.1145/2560018

78

1. INTRODUCTION

In the most recent years, with the advent of mobile Internet era, mobile phones have experienced an explosive development in both hardware and software and have been witnessing their revolutionary transformation from simple wireless communication tools to a powerful intelligent mobile platform. A variety of mobile phone based applications reply on the location information. For example, users can post on Facebook with a geo-location label. GeoLife envisions the service displaying shopping lists on a mobile phone when it is detected near a supermarket [Sohn et al. 2005].

One of the most popular technology enabling localization is to embed GPS devices in the mobile phone. While GPS works well in outdoor environments, it fails indoors, as

Authors' addresses: S. Li, Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07030; email: lishuai8@gmail.com; Y. Lou, School of Mechatronics and Information, Yiwu Industrial and Commercial College, Yiwu, Zhejiang, China; B. Liu, Department of Computer Science, University of Massachusetts, Amherst, MA 01003.

DOI: http://dx.doi.org/10.1145/2560018

the signal cannot penetrate constructions. This is especially the case, say in a tunnel or in an underground subway station, where even the communication signal is too weak to make any phone call, not to mention the GPS signal for localization. On the other hand, the relatively high cost of a GPS module also limits the popularization of GPS based localization to low cost mobile phones in current stage. Other alternatives, such as WiFi [Chen et al. 2006; Cheng et al. 2005] and GSM [Varshavsky et al. 2007] based localizations, require the predeployment of base stations with a known position and may therefore limit their applications.

Widely existing in most mobile phones, Bluetooth is originally designed for exchanging data with a low power consumption over short distances (the maximum communication range for Bluetooth is 100 meters, 10 meters, 5 meters for class 1, class 2, and class 3 transceivers, respectively.). Like all the other wireless signals, such as WiFi, GSM, etc., the signal strength at a given distance from the Bluetooth device varies due to propagation conditions, material coverage, antenna configurations and battery conditions [Tse and Viswanath 2005] and the calculated distance according to the received signal strength (RSS) often has a large error [Kaemarungsi and Krishnamurthy 2004]. Nevertheless, the nominal maximum range, which is measured under ideal conditions in open environments without obstacles along the signal propagation route, without material coverage, with a proper configuration of the antenna and with a full power of the battery, etc., gives an upper bound of the distance between two connected Bluetooth devices. Actually, with merely this proximity information and some additional ones, we are still able to localize the mobile phone with certain accuracy. In fact, a proportion of existing mobile phone users has the GPS functions with their phone, which may be utilized as beacons for those Bluetooth equipped but GPS nonexistent phones. In contrast to existing base station based localization strategy [Hay and Harle 2009; Pei et al. 2010], which requires a dense deployment of base stations to guarantee that each mobile phone is at least neighbored by three base stations [Doukhnitch et al. 2008], the proposed approach in this article thoroughly reduces the number of beacons by iteratively using already localized mobile phones as new beacons, as will be demonstrated in Section 5.

Mobility is an inherent property of the mobile phone network constructed by Bluetooth communication links, which distinguish the phone network from static networks, such as wireless sensor networks (WSNs). For WSNs, there is a similar class of localization strategy, which is called range-free localization [He et al. 2003]. The strategy in WSNs localizes sensors in a similar way based on the connectivity information of the network and the positions of beacon sensors. Since WSNs have invariant sensor positions, a time inefficient algorithm is still acceptable for WSN localization, which leaves living spaces for some unscalable algorithms [Doherty et al. 2001; Kulaib et al. 2011]. In contrast, a localization algorithm for mobile phones must be time-efficient relative to the moving speed of the phone. For example, an algorithm with a running time of one second results in an error more than 20 meters solely because of the mobility of a mobile phone user riding on a car with a speed of 50 miles/hour. Inspired by the great success of recurrent neural network on realtime signal processing [Skowronski and Harris 2007], robotics [Li et al. 2007, 2012], online optimization [Smith 1999], etc., we proposed a neural approach in Li et al. to tackle the problem in real time. In this article, we investigate this neural network approach in more detail both in theory and in simulation.

The remainder of this article is organized as follows. In Section 2, we formulate the Bluetooth aided mobile phone localization problem into a mathematic problem and present a recurrent neural network model to solve it. In Section 3, the convergence of the neural network is analyzed and it is proven to be convergent to a feasible solution of the problem. In Section 4, the hardware implementation of the neural network model

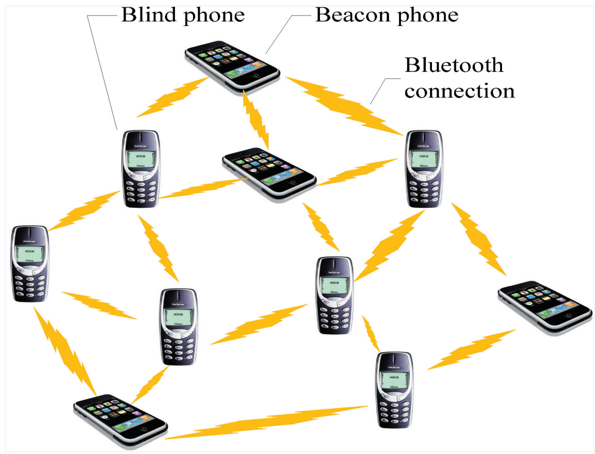

Fig. 1. Connectivity topology of the Bluetooth aided mobile phone localization system.

is explored. In Section 5, two applications are given and simulations are performed to demonstrate the effectiveness of our method. Section 6 concludes this article.

2. PROBLEM FORMULATION

For the convenience of problem formulation, we first give the following definitions.

Definition 2.1. Beacon phone: the mobile phone with a GPS device.

Definition 2.2. Blind phone: the mobile phone without GPS.

Both beacon phone and blind phone referred in this approach are assumed to be equipped with Bluetooth devices. Figure 1 sketches the connectivity topology of a phone network consisting of beacon phones and blind phones. In the network, the positions of beacon phones are obtained by GPS. Each Bluetooth connection link gives a constraint to the positions of mobile phones asides the link. In equation, we have

$$(x_i - x_j)^T (x_i - x_j) \leq R^2 \quad \text{for } i \in \mathbb{N}(j) \tag{1a}$$

$$x_k = \bar{x}_k \quad \text{for } k \in \mathbb{B}, \tag{1b}$$

where x_i, x_j represents the position of the ith and the jth mobile phone, respectively, R is the maximum communication range of the Bluetooth device, $\mathbb{N}(j)$ denotes the jth mobile phone's neighbor set, which includes all mobile phones connected to it via Bluetooth, $\mathbb{B}$ is the beacon phone set, $\bar{x}_k$ is the GPS measured position of the beacon phone labeled the kth.

Note that there is no explicit objective function but inequality and equality constraints in problem (1). The solution to this problem is generally not unique. Similar to applications like range-free localization of WSNs [Doherty et al. 2001; He et al. 2003] and communication connectivity maintenance in robot networks [Reich et al. 2011; Hsieh et al. 2008], etc., we are more concerned with finding a feasible solution in real time instead of finding all the feasible solutions. Based on this consideration, we explore finding a feasible solution to Problem (1) in real time via a recurrent dual neural network.

The solution of Problem (1) is identical to the one of the following normal optimization with an explicit objective function,

$$\text{minimize} \quad \sum_{i=1}^{n} \sum_{j \in \mathbb{N}(i)} w_{ij} \max\left\{(x_i - x_j)^T (x_i - x_j) - R^2, 0\right\}$$

$$\text{subject to} \quad x_k = \bar{x}_k \quad \text{for}\, k \in \mathbb{B}, \tag{2}$$

where n denotes the number of all mobile phones, $w_{ij} > 0$ is the weight of the connection between the ith and the jth phone. Note that the optimization problem (2) is non-smooth due to the presence of the function $\max(\cdot)$. The partial gradient relative to x_i (partial sub-gradient to be more exactly) of the objective function switches between $4\sum_{j\in\mathbb{N}(i)}(x_i - x_j)$ and 0 at the critical point $(x_i - x_j)^T(x_i - x_j) - R^2 = 0$. For smooth arbitration, we use the the following recurrent neural network, with the switching criteria augmented negative gradient evolution, to find a feasible solution of the optimization problem (2),

$$\dot{x}_i = -\epsilon \sum_{j \in \mathbb{N}(i)} w_{ij} I_{ij}(x_i - x_j), \tag{3}$$

where x_i is the position estimation of the blind mobile phone labeled i, $\epsilon > 0$ is a scaling factor, w_{ij} is a positive weight, I_{ij} is an indicator function defined as follows:

$$I_{ij} = \begin{cases} 1 & \text{if}\, (x_i - x_j)^T (x_i - x_j) - R^2 > 0 \\ 0 & \text{if}\, (x_i - x_j)^T (x_i - x_j) - R^2 \leq 0. \end{cases} \tag{4}$$

About the distributiveness of the proposed neural network model (3), we have the following remark.

Remark 2.3. The recurrent neural network (3) is a distributed one. Communication only happens between neighbor phones with direct Bluetooth connections. No routing or cross-hop communication is required for the implementation of the neural network (3). The distributed nature of the neural network thoroughly reduces the communication burden and makes the neural network scalable to a network with a large number of phones involved.

The following remark gives an intuitive interpretation of the working principle of the proposed neural network (3) for position estimation.

Remark 2.4. The dynamic evolution of x_i in the recurrent neural network (3) depends on its neighbor values x_j for $j \in \mathbb{N}(i)$. In detail, the neighbor phone x_j has an action $-\epsilon I_{ij}(x_i - x_j)$ on x_i. This action term is analogous to a force pointing from x_i to x_j and pulling x_i to x_j with an amplitude ϵ or 0 respectively when $\|x_i - x_j\| > R$ or $\|x_i - x_j\| \leq R$. This mechanism guides position estimations of neighbor phones to aggregate to within the maximum range R.

3. CONVERGENCE ANALYSIS

In this section, we study the convergence of the neural network (3) and the solution feasibility to the original problem (1). About this neural network, we have the following theorem,

THEOREM 3.1. *The recurrent neural network (3) with $\epsilon > 0$, w_{ij} for all possible i and j, asymptotically converges to a feasible solution x_i^* (for all i in the blind mobile phone set) of problem (1).*

PROOF. We construct the following Lyapunov function to analyze the dynamic of the recurrent neural network,

$$V = \sum_{i=1}^{n} \sum_{k \in \mathbb{N}(i)} w_{ik} \big((x_i - x_k)^T (x_i - x_k) - R^2 \big)^+, \tag{5}$$

where $x^+ = \max\{x\ 0\}$ for a scalar x. The derivative of $((x_i - x_k)^T (x_i - x_k) - R^2)^+$ is given by $\frac{\partial((x_i - x_k)^T (x_i - x_k) - R^2)^+}{\partial x_i} = 2I_{ik}(x_i - x_k)$ and $\frac{\partial((x_i - x_k)^T (x_i - x_k) - R^2)^+}{\partial x_k} = -2I_{ik}(x_i - x_k)$ respectively with I_{ik} representing the indicator function defined in (4).

The time derivative of V along the neural dynamic (3) is as follows,

$$\begin{aligned}
\dot{V} &= 2 \sum_{i=1}^{n} \sum_{k \in \mathbb{N}(i)} \big(w_{ij} I_{ik} (x_i - x_k)^T \dot{x}_i - w_{ij} I_{ik} (x_i - x_k)^T \dot{x}_k \big) \\
&= 2 \sum_{i=1}^{n} \left(\dot{x}_i^T \sum_{k \in \mathbb{N}(i)} w_{ik} I_{ik} (x_i - x_k) - \dot{x}_k^T \sum_{k \in \mathbb{N}(i)} w_{ik} I_{ik} (x_i - x_k) \right) \\
&= -2\epsilon \sum_{i=1}^{n} \left(\left\| \sum_{k \in \mathbb{N}(i)} w_{ik} I_{ik} (x_i - x_k) \right\|^2 + \left\| \sum_{k \in \mathbb{N}(i)} w_{ik} I_{ik} (x_i - x_k) \right\|^2 \right) \\
&= -4\epsilon \sum_{i=1}^{n} \left\| \sum_{k \in \mathbb{N}(i)} w_{ik} I_{ik} (x_i - x_k) \right\|^2 \\
&\leq 0,
\end{aligned} \tag{6}$$

where $\|\cdot\|$ represents the Euclidean norm of a vector. Following the procedure of LaSalle's Lemma [Khalil 2002], we let $\dot{V} = 0$ to find the largest invariant set,

$$\dot{V} = 0 \Rightarrow \sum_{k \in \mathbb{N}(i)} w_{ik} I_{ik} (x_i - x_k) = 0 \quad \text{for all } i. \tag{7}$$

The position x_i of the mobile phone i is a three dimensional vector in latitude/longitude/altitude (or simply two dimensional by assuming all positions have the same altitude). Considering the first dimension of vector x in (7), we have,

$$x_{i1} \sum_{k \in \mathbb{N}(i)} w_{ik} I_{ik} = \sum_{k \in \mathbb{N}(i)} w_{ik} I_{ik} x_{k1} \quad \text{for all } i, \tag{8}$$

where x_{i1}, x_{k1} represent the first dimension components of the vector x_{i1} and x_{k1} respectively. Defining that i_1^* represents the mobile phone with the largest value of x_{k1} for all k and i_2^* represents the second largest one and generally i_m^* represents the mth largest one, (8) yields the following for $i = i_1^*$,

$$x_{i_1^*1} \sum_{k \in \mathbb{N}(i_1^*)} w_{i_1^*k} I_{i_1^*k} = \sum_{k \in \mathbb{N}(i_1^*)} w_{i_1^*k} I_{i_1^*k} x_{k1}. \tag{9}$$

Note that $I_{ik} \geq 0$ for all i and k. Equation (9) means $x_{i_1^*1}$, which is the largest one in $\{x_{k1}\}$ for all k, can be represented as a weighted average of its neighbor values, that is, $x_{i_1^*1} = \sum_{k \in \mathbb{N}(i_1^*)} \frac{w_{i_1^*k} I_{i_1^*k}}{\sum_{k \in \mathbb{N}(i_1^*)} w_{i_1^*k} I_{i_1^*k}} x_{k1}$, if $\sum_{k \in \mathbb{N}(i_1^*)} w_{i_1^*k} I_{i_1^*k} \neq 0$. In fact, there is only one possibility that $x_{k1} = x_{i_1^*1}$ for $I_{i_1^*k} \neq 0$ in this case and otherwise it contradicts the fact that the strictly largest one cannot be expressed as a weighted average of the others. For those k satisfying $x_{k1} = x_{i_1^*1}$, x_{k1} is also the largest and letting $k = i_1^*$ yields similar conclusions. Recursively, we will finally conclude that $x_{k1} = x_{i1}$ for all i and k, which contradicts that the beacon phones are in different places. According to the above reasoning, we only need to consider $\sum_{k\in\mathbb{N}(i_1^*)} w_{i_1^*k} I_{i_1^*k} = 0$, which leads to $I_{i_1^*k} = 0$ for all $k \in \mathbb{N}(i_1^*)$. Let us then consider i_2^*, (8) yields the following for $i = i_2^*$,

$$x_{i_2^*1} \sum_{k\in\mathbb{N}(i_2^*)} w_{i_2^*k} I_{i_2^*k} = \sum_{k\in\mathbb{N}(i_2^*)} w_{i_2^*k} I_{i_2^*k} x_{k1} \text{ for all } k \in \mathbb{N}\left(i_2^*\right)$$

Note that $k = i_1^*$ is excluded as $I_{i_1^*k} = 0$ is valid for $k = i_2^*$ if i_1^* and i_2^* are neighbors. With a similar reasoning as for $i = i_1^*$, we conclude that $I_{i_2^*k} = 0$ for all $k \in \mathbb{N}(i_2^*)$. Following this procedure recursively with i_3^*, i_4^*, ..., we finally get $I_{ik} = 0$ for all possible i and k, which means $(x_i - x_j)^T(x_i - x_j) - R^2 \leq 0$ for $i \in \mathbb{N}(j)$ and all j. In other words, the largest invariant set coincide the solution to problem (1). According to LaSalle's Lemma [Khalil 2002], we concludes that the neural network asymptotically converges to a feasible solution of problem (1). This completes the proof. □

4. HARDWARE IMPLEMENTATION OF THE NEURAL NETWORK

The proposed neural network can either be implemented on microprocessors in series for the update of position estimation by discretizing the dynamic equation (3) or be implemented in analog circuits in parallel. In this section, we study the parallel implementation of the proposed model.

In the neural network (3), each blind mobile phone is associated with a dynamic neuron. We regard such a dynamic neuron as a module, which is a building block of the neural network. The modules interact with their neighbor modules and all the modules together perform the localization task and solve the problem (1). Different from conventional iterative methods, which may only be implementable in series, the proposed neural network can be implemented in analog circuits and accordingly processes signals in parallel and solve the problem in real time.

Figure 2 sketches the implementation architecture with analog devices of the neural module associated with the blind mobile phone labeled 1 (i.e., $i = 1$), where j_1, j_2, ..., j_k denote the neighbor phones (either blind phones or beacon phones). From Figure 2, we can see that summators, multipliers, linear amplifiers, nonlinear amplifiers (for the implementation of the indicator function) and integrators are employed in the implementation. The neural module gets input from modules associated with its neighbor phones and outputs its own position estimation. Equipped with such a hardware module, mobile phones are able to localize themselves with proximity information provided by the Bluetooth devices.

5. APPLICATIONS

In this section, two applications are considered and simulations are performed to show the effectiveness of the proposed approach for Bluetooth aided mobile phone localization in the corresponding applications.

5.1. Mobile Phone Localization in the Zhujiang Tunnel

The Zhujiang Tunnel (as shown in Figure 3) is a highway tunnel under the Pearl River in Guangzhou, China, with a total length of 1238.5 meters and a traffic of 10,000 vehicles per day on average. In the duration of traveling through the tunnel, drivers are not able to localize themselves due to the signal coverage by the constructions, which brings a lot of inconveniences for vehicle localization, especially in emergent

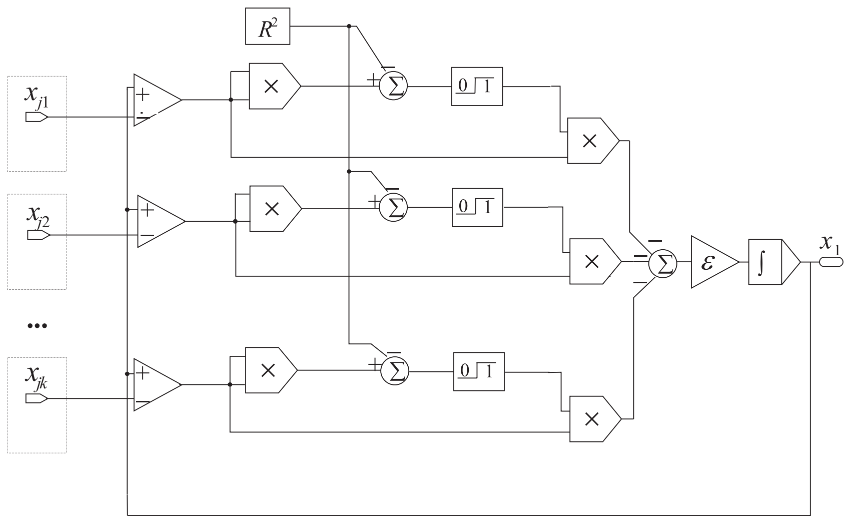

Fig. 2. Analog circuit architecture of a module in the proposed neural network.

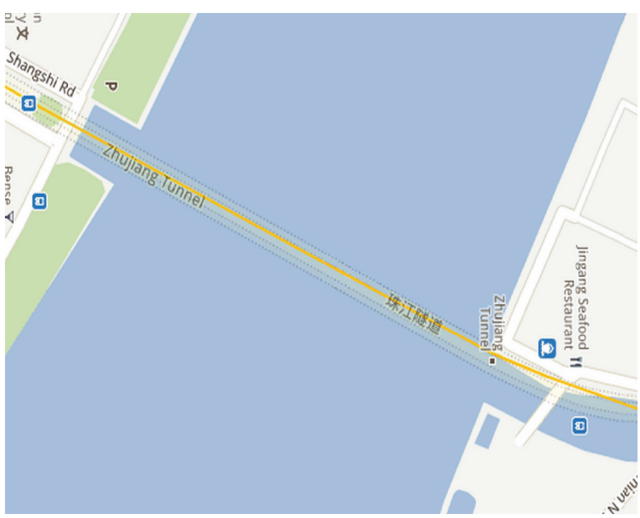

Fig. 3. The Zhujiang Tunnel under the Pearl River in Guangzhou, China.

situations, such as car accident. We use the Bluetooth aided mobile phone localization strategy to provide an option for rough localization of vehicles.

In the tunnel, there are three lanes for each direction and each lane has a width of 3.5 meters (as shown in Figure 4). We simplify the problem into a one dimensional problem as the width of the lane is much less than that of the communication radius (100 meters) as sketched in Figure 4. Note that the communication network is relatively stable for the phones on different vehicles in motion because the relative speed of the vehicles along the stream is very small even though each vehicle in motion has a high speed. It is the relative speed instead of the absolute one essentially influences the stability of the communication connections. In the real tunnel environment, there are several restrictions to the vehicle stream, such as no overtaking allowed, vehicles moving in similar speeds to maintain a constant intervehicle distance, etc. Due to

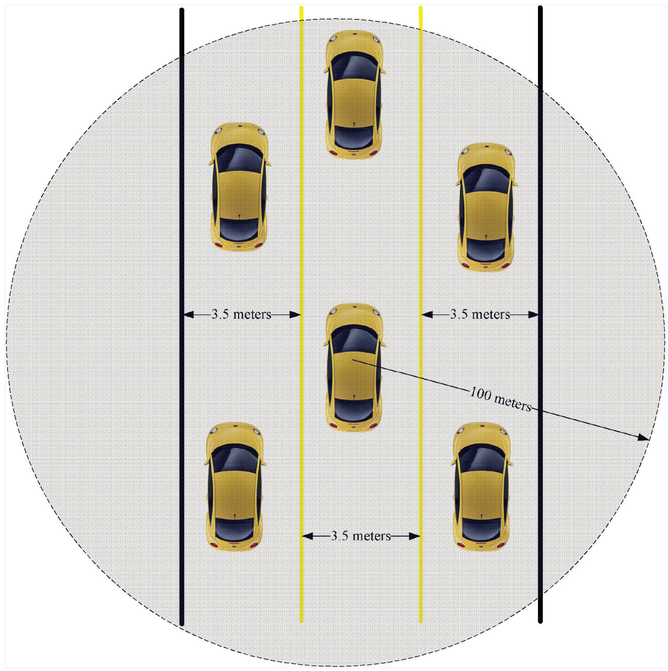

Fig. 4. The sketch of the Bluetooth aided localization in a tunnel.

these restrictions, the communication network is relatively stable and there is not that many switching on and off of connections, which may possibly introduce long connection latency.

We perform simulation experiment to validate the effectiveness of the proposed strategy for this particular application. Suppose there are 23 vehicles (the amount of vehicles is consistent with a rough estimation based on the length of the tunnel, the total traffic per day and an average speed of $60km/h$ for vehicles in the tunnel.) at certain time and they locate randomly along the tunnel, which is simplified to be a straight line (this is consistent with the observation from Figure 3). Mobile phones with GPS embedded inside cannot perform the self-localization task inside the tunnel, so we assume the phones inside the tunnel are all blind phones while the phones at the two ends of tunnel are possibly beacon phones. Particularly, we assume there are only two beacon phones, each of which locates at an end of the tunnel. The Bluetooth devices work in Class 3 transceiver mode with a maximum range of 100 meters. As we treat the tunnel as a straight line, the problem in nature is a one dimensional localization problem. For simulation convenience, we set the coordinates of the two beacon phones at 0 and 1238.5 meters, respectively. The values of x_i for all i is randomly initialized.

In the simulation, the neural network parameters $\epsilon = 10^5$ and $w_{ij} = 1$ for all possible i and j. A typical simulation result is shown in Figure 5. Since position estimations are randomly initialized, the initial estimation errors are very large. After running the simulation for 2×10^{-4} seconds, the neural network outputs an estimation, which is much closer to the true positions. The transient of position estimations is shown in Figure 6. To check the feasibility of the solution to problem (1), we use $\sum_{i=1}^{n} \sum_{j \in \mathbb{N}(i)} w_{ij} \max\{(x_i - x_j)^T (x_i - x_j) - R^2, 0\}$ to evaluate the performance. As shown in Figure 7, the value drops sharply with time and decays to zero at about 1.2×10^{-4}

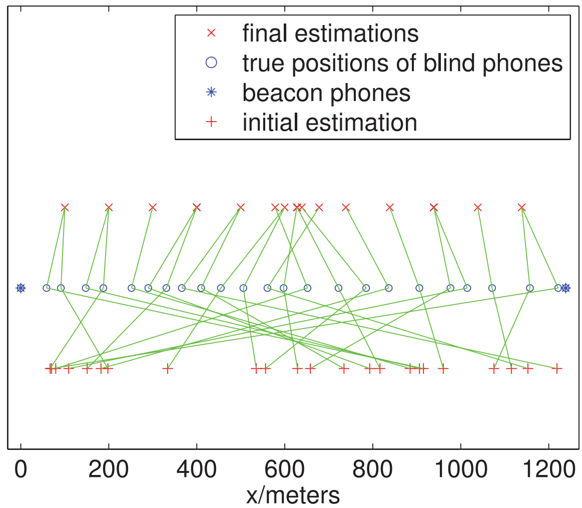

Fig. 5. Position estimation results in the tunnel localization application. In the figure, the lines connect the associated true phone positions, initially estimated positions and the finally estimated positions.

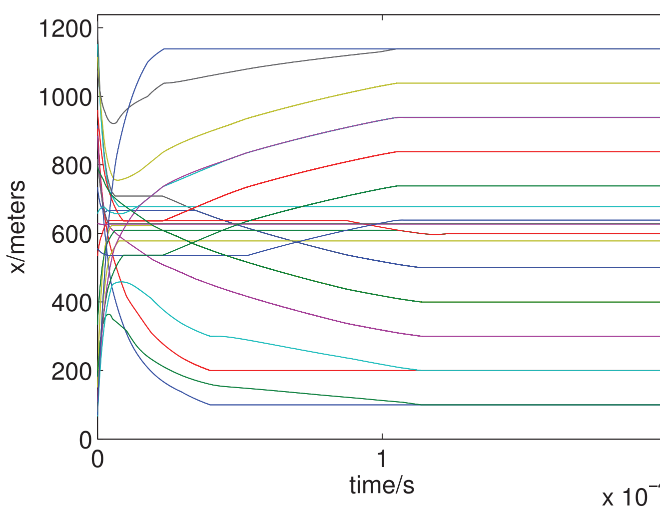

Fig. 6. Transient of the position estimation in the tunnel localization application.

seconds, which validates the effectiveness of the proposed method for solving problem (1) in real time in a tunnel environment.

5.2. Bluetooth Aided Localization in Supermarkets

Many supermarkets, such as Walmart, Bestbuy, etc., occupy a large area on a whole floor, and it is easy to get lost for customers shopping inside. This application aims to provide an option for indoor rough localizations in such a scenario without introducing extra devices.

In the simulation, the whole floor of the supermarket is assumed to be a 60×60 square meters area, and 121 customers with blind phones distribute in this area randomly and 9 beacon phones are deployed along the perimeter and at the center, with relative coordinates [0, 0], [30, 0], [60, 0], [60, 30], [60, 60], [30, 60], [0, 60], [0, 30], [30, 30]

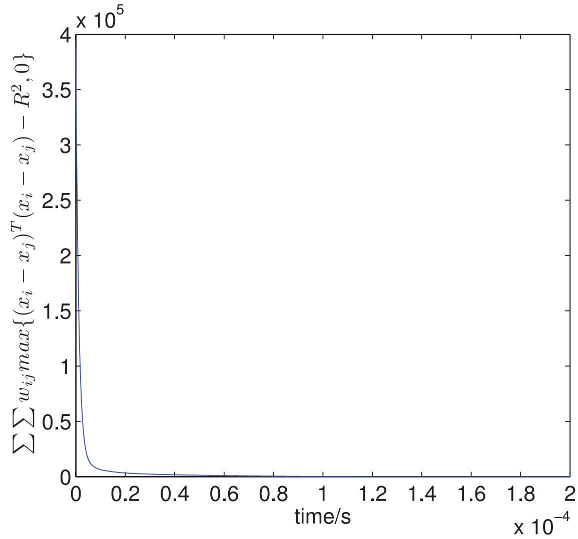

Fig. 7. The time evolution of $\sum_{i=1}^{n}\sum_{j\in\mathbb{N}(i)} w_{ij}\max\{(x_i - x_j)^T(x_i - x_j) - R^2, 0\}$ in the tunnel localization application.

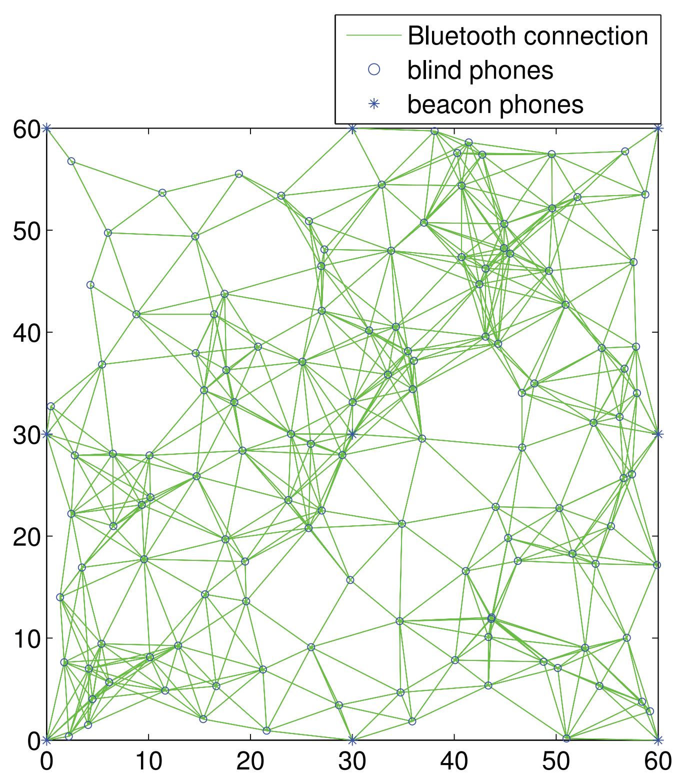

Fig. 8. True positions of phones in the Bluetooth network and the Bluetooth connection topology in the supermarket localization application.

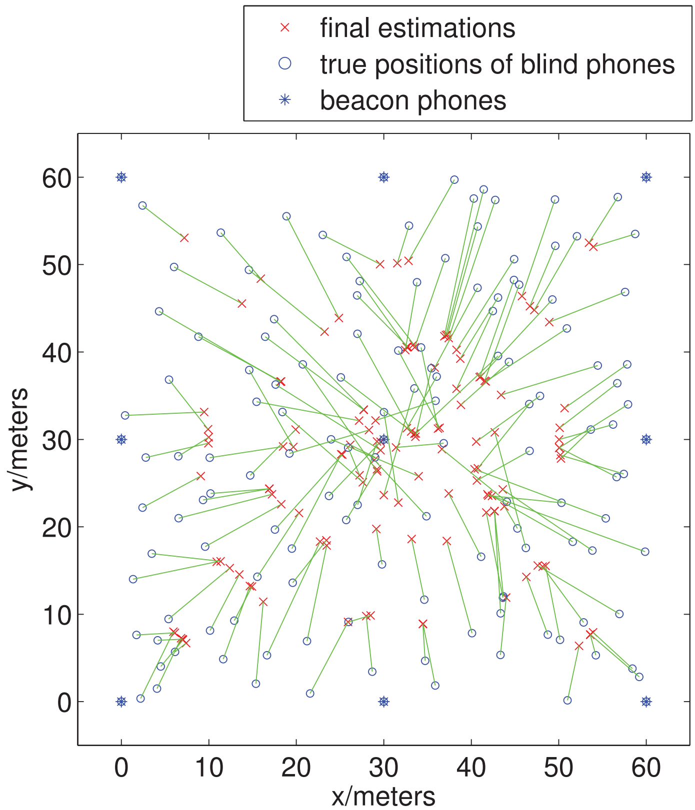

Fig. 9. Position estimation results in the supermarket localization application.

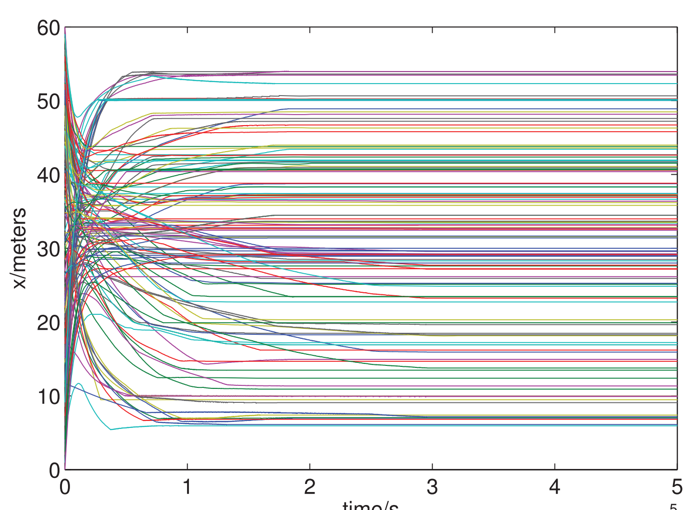

Fig. 10. Transient of the position estimation in x-direction in the supermarket localization application.

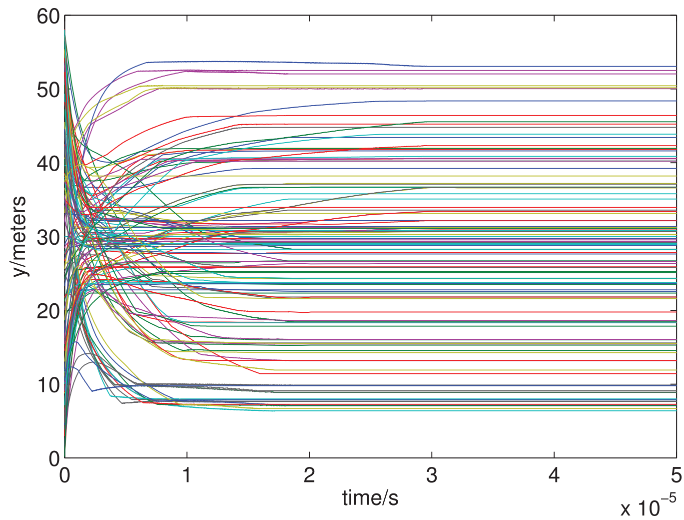

Fig. 11. Transient of the position estimation in y-direction in the supermarket localization application.

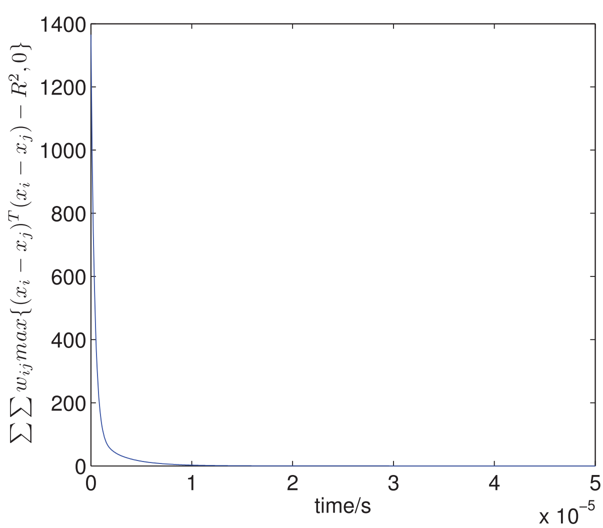

Fig. 12. Time evolution of $\sum_i \sum_j max\{(x_i - x_j)^T(x_i - x_j) - R^2, 0\}$ in the supermarket localization application.

respectively, as shown in Figure 8. The Bluetooth devices work in class 2 mode and have an maximum range of $R = 10$ meters. The scaling factor $\epsilon = 10^5$ and the connection weight w_{ij} equals 5 for connections with a beacon phones and 1 otherwise for the neural network. Figure 9 shows the estimated positions of blind phones by running the neural network for 5×10^{-5} seconds. The transient of estimated positions in x and y directions are plotted in Figure 10 and Figure 11, respectively. Figure 12 shows the evolution of $\sum_i \sum_j max\{(x_i - x_j)^T(x_i - x_j) - R^2, 0\}$ with time, which is an quantitive evaluation of the feasibility of the solution to the mobile phone localization problem (1). The value starts from 1365.7 and drops to 0 at the end of the simulation, which demonstrates the effectiveness of the proposed approach.

Table I.
Comparisons of the AB method and our method on the performances under different parameter setups.

Parameters			Performances				
AB method vs. our method	No. of beacons	No. of blind phones	E_1	E_2	Localization rate	PC time (seconds)	Theoretical time (seconds)
AB method	9	110	*	*	0%	*	*
Our method	9	110	0.0019	10.3812	100%	2.1513	5×10^{-5}
AB method	9	130	*	*	0%	*	*
Our method	9	130	0.0021	10.3123	100%	2.9175	5×10^{-5}
AB method	9	150	*	*	0%	*	*
Our method	9	150	0.0027	9.7518	100%	3.8103	5×10^{-5}
AB method	9	170	*	*	0%	*	*
Our method	9	170	0.0034	8.4437	100%	5.3992	5×10^{-5}
AB method	16	110	*	*	0%	*	*
Our method	16	110	0.0023	7.5081	100%	2.3054	5×10^{-5}
AB method	16	130	*	*	0%	*	*
Our method	16	130	0.0031	6.6802	100%	3.1723	5×10^{-5}
AB method	16	150	*	*	0%	*	*
Our method	16	150	0.0026	6.4727	100%	4.2393	5×10^{-5}
AB method	16	170	*	*	0%	*	*
Our method	16	170	0.0029	5.8742	100%	5.3992	5×10^{-5}
AB method	25	110	*	*	0%	*	*
Our method	25	110	0.0019	5.5040	100%	2.7278	5×10^{-5}
AB method	25	130	*	*	0%	*	*
Our method	25	130	0.0029	5.4326	100%	3.9657	5×10^{-5}
AB method	25	150	*	*	0%	*	*
Our method	25	150	0.0027	5.3977	100%	5.2791	5×10^{-5}
AB method	25	170	*	*	0%	*	*
Our method	25	170	0.0021	5.3054	100%	8.0778	5×10^{-5}
AB method	36	110	*	*	17.27%	*	*
Our method	36	110	0.0022	4.7553	100%	3.4620	5×10^{-5}
AB method	36	130	*	*	20.00%	*	*
Our method	36	130	0.0031	4.3626	100%	4.1029	5×10^{-5}
AB method	36	150	*	*	24.12%	*	*
Our method	36	150	0.0024	4.3245	100%	5.7575	5×10^{-5}
AB method	36	170	*	*	24.67%	*	*
Our method	36	170	0.0029	4.0440	100%	8.0111	5×10^{-5}

Table I shows the comparisons of the AB method (anchor-based Bluetooth localization method) [Raghavan et al. 2010; Feldmann et al. 2003; Keiser et al. 2006] and our method on the performances under different parameter setups. Based on the value of RSS, The AB method uses trilateration to estimate the location of a Bluetooth device with connections to at least three beacon phones. To compare the AB method with the proposed strategy, we consider the cases with different number of beacon phones and different number of blind phones. The simulation is performed with the programming language Matlab 7.8 on a laptop with the Intel (R) Core(TM) 2 Duo CPU at 1.80 GHz and 2GB of RAM. Note that the simulation program performs the localization algorithms for all the beacon phones and all the blind phones. In real application, the localization algorithm will be run in a distributed manner separately by all phones. In the simulation, the beacon phones are deployed uniformly in the area and the results are averaged based on the data collected in 50 Monte Carlo runs with random initializations. In the table, '*' means the performance index is meaningless as in the simulated scenario, the AB method fails and most blind phones cannot be localized by using the AB method with such a limited number of beacon phones. We use the localization rate, which is defined as the ratio between the localized blind phones and the total number of all blind phones, as a measure of the performance. As observed in the table, the localization rate of the AB method is 0% for the cases with beacon phones less than or equal to 25. Even for the case with 36 beacon phones and 170 blind phones, the localization rate is only 24.67%, which means about 75% blind phones cannot be localized by the AB method. In contrast, for all the simulated cases, the localization rate of our method is always 100%, meaning that all blind phones can be localized by our method. This result demonstrates the advantage of our method in environments with sparse deployment of beacons. The underlying reason can be intuitively explained, as the AB method fails to localize a blind phone, say phone A, if there are fewer than three beacon phones connected to it, no matter how many blind phones connected to the blind phone A. Differently, our method still works well in such a scenario. It is worth noting that the theoretical running time of the proposed algorithm is much less than the PC time (the CPU time in the simulation). This is because, on one hand, the simulation program simulates all phones on a single computer while the algorithm is expected to run separately on all phones in parallel in real application and on the other hand the neural network implementation in analog circuits can complete the computation when the neural evolution converges. As to the software implementation of the neural network model, the running time can be estimated by the ratio between the PC time listed in the table and the total number of phones simulated, whose value is still acceptable for real-time processing. Moreover, the localization error E_1, defined as $E_1 = \sum_{i=1}^{n} \sum_{j \in \mathbb{N}_i} max\{(x_i - x_j)^T (x_i - x_j) - R^2, 0\}$ with x_i denoting the estimated position of the ith phone, and the localization error E_2, defined as $E_2 = \sqrt{\sum_{i=1}^{n} (x_i - x_{ri})^T (x_i - x_{ri})/n}$ with x_{ri} denoting the real position of the ith blind phone, are both shown in Table I, from which we can see that E_1 is very close to zero for our method in all cases as $E_1 = 0$ is identical to the expression (1) and E_2 decreases with the increase of the number of beacons and also decreases with the increase of the number of blind phones.

6. CONCLUSIONS

In this article, we proposed a recurrent neural network based method for Bluetooth aided mobile phone rough localization. The problem is abstracted to solve a set of inequalities defined on a Bluetooth connection network and a recurrent neural network is proposed to solve the problem in real time. The convergence of the proposed neural network and the feasibility of the neural solution are proven in theory. The architecture of the circuit implementation of the neural network is given. Finally, applications of the method to localization of drivers in a tunnel and customer localization in a supermarket are explored and simulated. Simulations demonstrate effectiveness of the method.

REFERENCES

Mike Y. Chen, Timothy Sohn, Dmitri Chmelev, Dirk Haehnel, Jeffrey Hightower, Jeff Hughes, Anthony Lamarca, Fred Potter, Ian Smith, and Alex Varshavsky. 2006. Practical metropolitan-scale positioning for gsm phones. In *Proceedings of the 8th International Conference on Ubiquitous Computing*. Springer, 225–242.

Y. C. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm. 2005. Accuracy characterization for metropolitan-scale wi-fi localization. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services (MobiSys'05)*. ACM, New York, 233–245.

L. Doherty, K. S. J. Pister, and E. L. Ghaoui. 2001. Convex position estimation in wireless sensor networks. In *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'01)*. Vol. 3, 1655–1663.

E. Doukhnitch, M. Salamah, and E. Ozen. 2008. An efficient approach for trilateration in 3D positioning. *Comput. Commun.* 31, 4124–4129.

S. Feldmann, K. Kyamakya, A. Zapater, and Z. Lue. 2003. An indoor bluetooth-based positioning system: Concept, implementation and experimental evaluation. In *Proceedings of the International Conference on Wireless Networks*. 109–113.

Simon Hay and Robert Harle. 2009. Bluetooth tracking without discoverability. In *Location and Context Awareness*, T. Choudhury, A. Quigley, T. Strang, and K. Suginuma, Eds., Lecture Notes in Computer Science, vol. 5561, Springer, 120–137.

T. He, C. Huang, Brian M. Blum, J. A. Stankovic, and T. Abdelzaher. 2003. Range-free localization schemes for large scale sensor networks. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom'03)*. ACM, New York, 81–95.

M. A. Hsieh, A. Cowley, V. Kumar, and C. J. Taylor. 2008. Maintaining network connectivity and performance in robot teams. *J. Field Rob.* 25, 1–2, 111–131.

K. Kaemarungsi and P. Krishnamurthy. 2004. Properties of indoor received signal strength for wlan location fingerprinting. In *Proceedings of the 1st Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MOBIQUITOUS'04)*. 14–23.

Oliver Keiser, Philipp Sommer, and Bernhard Tellenbach. 2006. Bluelocation ii: A localization infrastructure for bluetooth enabled mobile devices. Tech Rep., Swiss Federal Institute of Technology.

H. Khalil. 2002. *Nonlinear Systems*. Prentice Hall.

A. R. Kulaib, R. M. Shubair, M. A. Al-Qutayri, and J. W. P. Ng. 2011. An overview of localization techniques for wireless sensor networks. In *Proceedings of the International Conference on Innovations in Information Technology*. 167–172.

S. Li, S. Chen, B. Liu, Y. Li, and Y. Liang. 2012. Decentralized kinematic control of a class of collaborative redundant manipulators via recurrent neural networks. *Neurocomputing*.

S. Li, S. Chen, Y. Lou, B. Lu, and Y. Liang. A recurrent neural network for inter-localization of mobile phones. In *Proceedings of the IEEE International Joint Conference on Neural Networks*.

S. Li, M. Q. H. Meng, and W. Chen. 2007. Sp-nn: A novel neural network approach for path planning. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics*. 1355–1360.

Ling Pei, Ruizhi Chen, Jingbin Liu, Tomi Tenhunen, Heidi Kuusniemi, and Yuwei Chen. 2010. Inquiry-based bluetooth indoor positioning via RSSI probability distributions. In *Proceedings of the 2nd International Conference on Advances in Satellite and Space Communications (SPACOMM'10)*. IEEE, 151–156.

Aswin N. Raghavan, Harini Ananthapadmanaban, Manimaran Sivasamy Sivamurugan, and Balaraman Ravindran. 2010. Accurate mobile robot localization in indoor environments using bluetooth. In *Proceedings of the IEEE International Conference on Robotics and Automation* (*ICRA'10)*. 4391–4396.

J. Reich, V. Misra, D. Rubenstein, and G. Zussman. 2011. Connectivity maintenance in mobile wireless networks via constrained mobility. In *Proceedings of the Annual Joint Conference of the IEEE Computer and Communications Societies*. 927–935.

M. D. Skowronski and J. G. Harris. 2007. Noise-robust automatic speech recognition using a predictive echo state network. *IEEE Trans. Audio Speech Lang. Process.* 15, 5, 1724–1730.

K. A. Smith. 1999. Neural networks for combinatorial optimization: a review of more than a decade of research. *INFORMS J. Comput.* 11, 15–34.

T. Sohn, K. A. Li, G. Lee, I. E. Smith, J. Scott, and W. G. Griswold. 2005. Place-its: A study of location-based reminders on mobile phones. In *Proceedings of the ACM International Conference on Ubiquitous Computing*. M. Beigl, S. Intille, J. Rekimoto, and H. Tokuda, Eds., Lecture Notes in Computer Science, vol. 3660, Springer, 232–250.

D. Tse and P. Viswanath. 2005. *Fundamentals of Wireless Communication*. Cambridge University Press, Cambridge, UK.

A. Varshavsky, E. Lara, J. Hightower, A. LaMarca, and V. Otsason. 2007. GSM indoor localization. *Pervasive Mob. Comput.* 3, 698–720.

Received November 2011; revised February 2012; accepted May 2012