



Brief paper

Cellphone geolocation via magnetic mapping[☆]Yan Cui, Ruxiao An, Kartik B. Ariyur¹

School of Mechanical Engineering, Purdue University, West Lafayette, IN, 47907, USA

ARTICLE INFO

Article history:

Received 15 October 2013

Received in revised form

23 September 2014

Accepted 13 October 2014

Available online 18 November 2014

Keywords:

Position location

Magnetic fields

Estimation algorithms

Interval analysis

Multisensor integration

ABSTRACT

We develop novel algorithms based on interval analysis with theoretical guarantees, to augment the accuracy of cell phone geolocation by taking advantage of local variations of magnetic intensity. Thus, the sources of disturbances to magnetometer readings caused indoors are effectively used as beacons for localization. We construct a magnetic intensity map for an indoor environment by collecting magnetic field data over each floor tile. We then test the algorithms without position initialization and obtain indoor geolocation to within 2 m while slowly walking over a complex path of 80 m. The geolocation errors are smaller in the vicinity of large magnetic disturbances. Finally, we fuse the magnetometer measurement with inertial measurements on the cell phone to yield even smaller geolocation errors of under 50 cm for a moving user. Our theoretical results connect geolocation accuracy to combinations of sensor and map properties.

© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

Global Positioning System (GPS) receivers on cell phones offer accurate geolocation for customers, but these fail to be reliable in areas subject to multipath interference, shadowing and occlusion of a line of light to the satellites (Bajaj, Ranaweera, & Agrawal, 2002; Byun, Hajj, & Young, 2001). These GPS-denied areas include urban and natural canyons, forests, and indoor locations, or in the case that users are reluctant to drain battery power switching on GPS. The accuracy of cell phone geolocation in these areas with poor GPS performance is demanded by customers, especially for the indoors and urban canyon areas in which they work daily. Geolocation of any cell phone is *a priori* obtained to be within a range through the nearest cell tower (Weiss, 2003). To obtain a more accurate estimate, several approaches have been developed over the years. One main approach is to use the communication between the phone and the transceiver tower, including received signal strength (RSS) (Martin, Vinyals, Friedland, & Bajcsy, 2010; Rap-

paport, 1996), the time of arrival (TOA) (Bshara, Orguner, Gustafsson, & Van Biesen, 2010; Qi & Kobayashi, 2003) and the time difference of arrival (TDOA) (Sayed & Tarighat, 2005). Another approach using local landmarks includes utilizing indoor Wi-Fi transceivers and radio frequency identification (RFID) (Bekkali, Sanson, & Matsumoto, 2007), which performs better indoors. Moreover, by using the embedded sensors (Fang, Lin, & Lee, 2008) on cell phones, algorithms utilizing inertial measurement units (IMUs) and magnetometers have been introduced. Many previously developed algorithms and applications are recapitulated and compared in Gustafsson and Gunnarsson (2005).

Local fluctuations in magnetic field intensity make it possible to augment these methods for geolocation in some GPS-denied areas. With the measurement from magnetometers, a magnetic intensity map can be constructed by collecting measurements at each of the intersections on the meshed grid. In the case of indoor navigation using magnetometers (Bonnet & Héliot, 2007; Chung et al., 2011; Haverinen & Kemppainen, 2009), the mapping function can match each reading from a sensor to an area in a meshed grid from a floor plan. In our research, we use two types of magnetometers to build up the magnetic intensity map by collecting the steady-state value of the outputs at each point. We then develop algorithms based on interval analysis (Gning, Abdallah, & Bonnifait, 2007; Jaulin, Kieffer, Didrit, & Walter, 2001; Moore, Kearfott, & Cloud, 2009; Schweppe, 1973) to estimate each single measurement in quasi-static estimation and track the path of the cell phone user. For a single measurement, we develop an optimization algorithm based on concepts from Jaulin et al. (2001) to bound the local estimation into small 2-D intervals, and evaluate the performance by showing

[☆] This work was funded by Google Inc. under grant number #301307. The material in this paper was partially presented at the 25th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2012), September 17–21, 2012, Nashville, Tennessee, USA. This paper was recommended for publication in revised form by Associate Editor Antonio Vicino under the direction of Editor Torsten Söderström.

E-mail addresses: cui4@purdue.edu (Y. Cui), an23@purdue.edu (R. An), kariyur@purdue.edu (K.B. Ariyur).

¹ Tel.: +1 765 494 8613.

the relationship between sharpness of map, quality of sensor, and the geolocation error. To track the path of a walking user, we develop a dynamic algorithm to fuse each of the quasi-static interval estimates and generate a path of geolocation intervals with reduction of estimate size.

Moreover, with the help of embedded IMUs on smart phones, some auxiliary inertial navigation systems have been introduced (Bekkali et al., 2007; Cui & Ariyur, 2011; Fang et al., 2005, 2008; Jirawimut et al., 2003). The three-axis inertial measurements offer accelerations and angular rates at 16 Hz, with the advantage of introducing detailed information of the dynamic model, which can potentially improve the performance of the magnetic map navigation over short intervals of time. Hence, we develop an algorithm to fuse the IMU with the magnetometer, including an inertial update phase and a magnetometer update phase, in which estimated intervals being updated by inertial measurements are intersected with the next estimate at any sampling time. The result from dynamic estimation via magnetic map is improved significantly reducing the geolocation error. We then compare the result with the particle filter based on Monte-Carlo simulation (MCS) (Arulampalam et al., 2002; Doucet & Johansen, 2008), which is another algorithm for the grid-based tracking and sensor fusion. We apply the theory of particle filter into the tracking system to fuse the IMU with magnetometer and compare the performance of both algorithms, and the greatest precision is obtained from the interval analysis with estimate error less than 50 cm.

We organize this paper as follows, the system model and magnetic intensity map construction are introduced in Section 2. In Section 3, we introduce basic concepts from interval analysis as the tool to solve the problem. We present algorithms, theorems and results of both quasi-static estimation and dynamic estimation in Section 4. We then show the fusion process with IMU and comparison with particle filter in Section 5, and draw conclusions in Section 6.

2. Magnetic intensity map

The magnetic intensity map includes meshed grids and interaction nodes on it. Each node has the data collected from magnetometer and the position marked on the geometric map. Suppose we have a plane region $\mathbb{D} \subset \mathbb{R}^2$ considered as the *a priori* region, then any node in the region has a unique magnetic intensity. The value depends on the macro geolocation of this point obtained from the standard International Geomagnetic Reference Field (IGRF) (Finlay et al., 2010), and the impact from local extremely low frequency bias such as iron structure and devices (Burnett & Yaping, 2002). We define our mapping function as follows:

Definition 1 (*Mapping Function*). The local magnetic intensity map is defined as the function from a point $(x, y) \in \mathbb{D} \subset \mathbb{R}^2$ to a magnetic intensity value m , $m : \mathbb{R}^2 \rightarrow \mathbb{R}^1$.

$$\forall (x, y) \in \mathbb{D}, m(x, y) = |(m_I(x, y) + m_B(x, y))|. \quad (1)$$

The $R_{3 \times 3}$ rotation matrix has the property of $|R_{3 \times 3}| = 1$ and eliminates the rotation effects when calculating the norm value. The map set including all of magnetic intensity values $m(x, y)$ is then defined as M . $M = \{m(x, y) \mid (x, y) \in \mathbb{D}\}$. In our case, we use a 3-axis magnetometer to measure $(m_x, m_y, m_z)^T$ and take its norm value $|m|$ to build up the map and hence we have:

$$m(x, y) = |(m_I + m_B)| = \sqrt{m_x^2 + m_y^2 + m_z^2}. \quad (2)$$

So far we have the model to construct the map. However, we need to consider the measurement with noise and we have:

$$z_m = \sqrt{m_x^2 + m_y^2 + m_z^2} + v_m \quad (3)$$

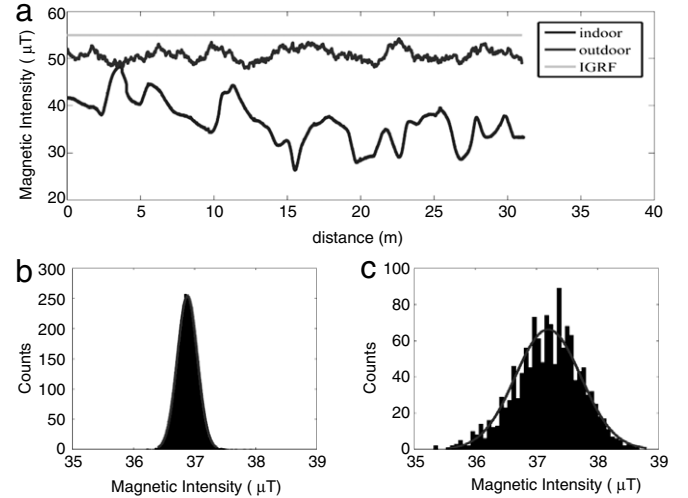


Fig. 1. Example of the magnetic intensity measurement. (a) The plot of local magnetic intensity from IGRF, outdoors, and indoors over 30 m. (b) The histogram of static measurement from 3DM-GX3 sensor, and $\mu_{3DM} = 35.9407$; $\sigma_{3DM}^2 = 0.0066$. (c) The histogram of static measurement from cell phone sensor, and $\mu_{phone} = 36.0952$; $\sigma_{phone}^2 = 0.0722$.

where z_m is the measurement and v_m is the measurement noise. An example of the measurement of magnetic intensity is shown in Fig. 1, including local magnetic intensities from the IGRF model, an outdoor test around the ME building on Purdue campus, and an indoor test inside the ME building. From Fig. 1(a), the magnetic intensity fluctuates more indoors than outdoors, or the IGRF model, which permits us to use the larger fluctuations effectively as beacons. The noise in this case is non-central χ distributed with 3 degrees of freedom. This is because the magnetic vector measurement has Gaussian noise. In our calculations however, we only use an empirical standard deviation of this noise to construct intervals of uncertainty from our measurements.

We build up the map experimentally by collecting data on each node, and obtain the mean value after 2 s of measurement. The distance between two grid lines is 0.3 m. We place devices on a plastic cart which was tested to have no influence on the magnetic intensity measurement. The floor plan and the experimental region are shown in Fig. 2. An example of the magnetic intensity map is shown in Fig. 5(b). Having built up the map, we need to design algorithms to match the real measurement z_m to the map and estimate the geolocation. Here we define the cost function $c(x, y)$ as the error between a single measurement z_m at a random point (x, y) , and the magnetic intensity $m(x, y)$ on the map.

$$c(x, y) = |z_m - m(x, y)|. \quad (4)$$

Hence, we need to find the local minimizers that minimize $c(x, y)$. Then we insert those local minimizers into an algorithm and approximately find the global minimizer which yields optimal geolocation. This is done with the mathematics of intervals which is introduced in the following section.

3. Interval analysis

Interval Analysis (IA) yields sets as estimates of optima or system states in sensor fusion (Horn et al., 2003; Jaulin et al., 2001; Jaulin & Walter, 1993; Kieffer, Jaulin, Walter, & Meizel, 2000; Moore et al., 2009). We use the initial magnetometer reading to determine all the possible spatial intervals that can contain the measurement, and then intersect them over subsequent readings to eliminate all local minimizers of $c(x, y)$. Interval analysis is better suited than the Kalman filter or the particle filter for this estimation task that exploits a random field, rather than being thrown off

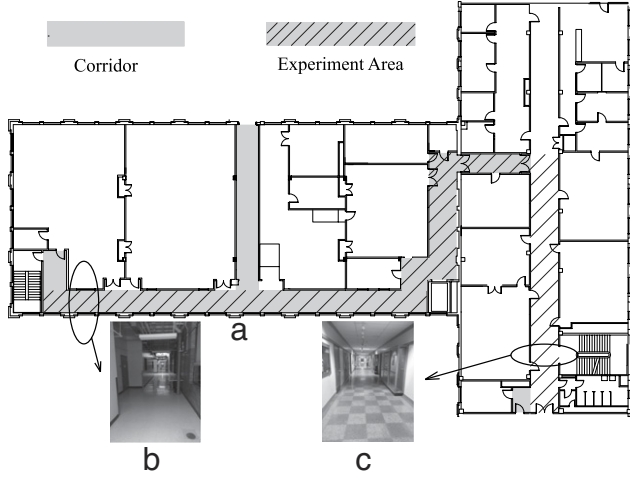


Fig. 2. Experiment area in ME Building: (a) floor plan of the second floor and we take measurements on the shaded area; (b), (c) the snapshot of two ends of the corridor, having positions circled on the floor plan.

by it. This is because a Kalman or particle filter assumes stationary process noise, which does not hold in our problem. We summarize the basic definitions and concepts necessary for our use of magnetic maps. Define an interval on one dimension \mathbb{R} , noted as $[x]$. Then the lower bound $lb(x) = \underline{x}$ and upper bound $ub(x) = \bar{x}$ and the width are defined as:

$$\underline{x} \triangleq \sup\{a \in \mathbb{R} \cup \{-\infty, \infty\} \mid \forall x \in [x], a \leq x\} \quad (5)$$

$$\bar{x} \triangleq \inf\{b \in \mathbb{R} \cup \{-\infty, \infty\} \mid \forall x \in [x], x \leq b\} \quad (6)$$

$$w([x]) = \bar{x} - \underline{x}. \quad (7)$$

The four classical operations of real arithmetic, namely addition, subtraction, multiplication and division can be extended to intervals (Jaulin et al., 2001). For any such binary operator denoted by \diamond , performing the operation \diamond on the intervals $[x]$ and $[y]$ means computing:

$$[x] \diamond [y] = \{x, y \in \mathbb{R} \mid x \in [x], y \in [y]\}. \quad (8)$$

Next, two basic operations, intersection and union are applied to intervals, which are useful in the following algorithms.

$$[x] \cap [y] = \{z \in \mathbb{R} \mid z \in [x] \text{ and } z \in [y]\} \quad (9)$$

$$[x] \cup [y] = \{z \in \mathbb{R} \mid z \in [x] \text{ or } z \in [y]\}. \quad (10)$$

In this paper, we apply the interval analysis on the 2-D estimation, which gives the set-up of an interval set for a certain area. Hence, we quote the definition of *interval real vector* for convenience.

Definition 2 (Interval Real Vector (Jaulin et al., 2001)). An *interval real vector* $[x]$ is a subset of \mathbb{R}^n that can be defined as the Cartesian product of n closed intervals. When there is no ambiguity, $[x]$ will simply be called an interval vector, or a *box*. It is written as:

$$[x] = [x_1] \times [x_2] \times \cdots \times [x_n], \quad (11)$$

$$[x_i] = [\underline{x}_i, \bar{x}_i] \quad \text{for } i = 1, \dots, n.$$

Hence, non-empty boxes are axis-aligned, and the case $n = 2$ is what we consider in this work. A non-empty box $[p] \subset \mathbb{R}^2$ is defined as:

$$[p] = [x] \times [y]. \quad (12)$$

Recall (7) and the width vector of a box on \mathbb{R}^2 is

$$\begin{aligned} \mathbf{w}([p]) &= (w_x([p]), w_y([p]))^T \\ &= (w([x]), w([y]))^T. \end{aligned} \quad (13)$$

When we use interval estimates, we need to consider functions of intervals. Consider two sets X and Y and a function: $f : X \rightarrow Y$. If $X_1 \subset X$, the function of X_1 is:

$$f(X_1) = \{f(x) \mid x \in X_1\} \quad (14)$$

and this is useful when using cost function equation (4) in interval analysis.

The definitions and properties of intervals are used in quasi-static estimation and the fusion with IMU, obtaining rectangular axis-aligned 2-D intervals rather than irregular shaped estimates, which will be discussed in the next sections.

4. Methods and algorithms

Now that we have developed the basic concepts of the magnetic intensity map and presented the needed tools from interval analysis, we then introduce algorithms of quasi-static estimation and dynamic estimation based on single and multiple measurements respectively. The structure of the estimation procedure is presented in Fig. 3. Having the map constructed with meshed floor plan and data collected on each node, every single measurement is processed via quasi-static estimation to obtain results, from which dynamic estimation updates and generates an estimate path to minimize the estimate error.

4.1. Quasi-static estimation (QSE)

For any single measurement, we want to obtain the optimal estimate. In our case, we developed an algorithm to bound the geolocation in 2-D intervals. Recall the cost function equation (4), we are interested in the geolocation that minimize the cost function in intervals. Referring to the definition of a function over an interval from Eq. (14), we convert the cost function into a 2-D form.

Definition 3 (Cost Function Interval). Suppose we have a 2-D box $[p] \subset \mathbb{R}^2$, and recall equations (4) and (14), the cost function over a box $[p]$ is defined as $c : \mathbb{R}^2 \rightarrow \mathbb{R}$:

$$c([p]) = \{c(x, y) \mid (x, y) \in [p]\}. \quad (15)$$

Then the cost function interval for an interval in 2-D $[c]([p])$ is defined as:

$$[c]([p]) = [\underline{c}([p]), \bar{c}([p])]. \quad (16)$$

Let $[p_i]$ be the interval including local minimizer of the cost function $c([p])$, the QSE process becomes an optimization problem:

$$[\hat{p}_i] = \arg \min_{(x,y) \in [p]} c([p]). \quad (17)$$

Any time collecting a measurement z_m , we start scanning all nodes on the map and evaluate the cost function in the rough region $D \subset \mathbb{R}^2$. As shown in Fig. 1, the magnetic intensity measurement has error when collecting data, we are not able to find the point which exactly makes the cost value equal to zero. Hence, a threshold value $\bar{c} > 0$ is defined as a judgment to bound the local minimizers into intervals.

Remark 1. If $\inf\{[c]([p])\} < \bar{c}$, then there exists a point $(x_i, y_i) \in [p]$ such that $c(x_i, y_i) < \bar{c}$.

Remark 2. For a small value $\bar{c} > 0$, $\bar{c} \ll 1$ that has

$$\inf\{[c]([\hat{p}_i])\} < \bar{c} \quad (18)$$

then the local minimizer $(x_0, y_0) \in [\hat{p}_i]$.

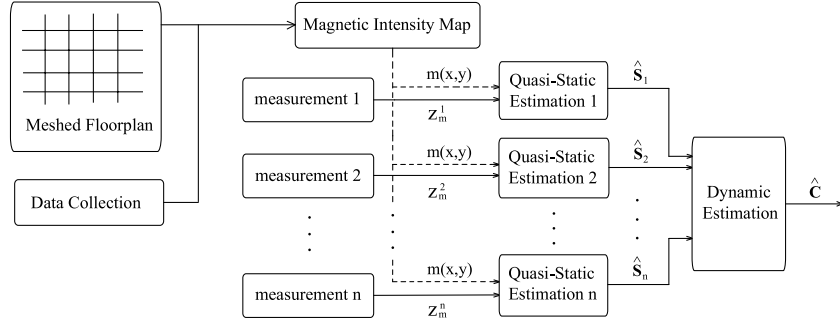


Fig. 3. System structure of using magnetic intensity map.

Table 1
Algorithm of quasi-static estimation.

Algorithm 1: quasi-static estimation

1. Initialize the wrapper set $[p]$: $B = \{([p], \infty)\}$
2. **while** $B \neq \emptyset$ % iterations
3. $([p], [c]([p])) = B(i)$;
4. **if** $\inf\{[c]([p])\} \geq \bar{c}$
5. remove the pair $([p], [c]([p]))$ from B
6. **else**
7. **if** $w([p]) < \epsilon$, put $[p]$ into \hat{S}
8. **else**
9. bisect $[p]$ into $[p_1], [p_2]$
10. $B \leftarrow ([p_1], [c]([p_1])), ([p_2], [c]([p_2]))$
11. **end if**
12. **end if**
13. $i++$
14. **end while**

Remarks 1 and 2 give the intuition of obtaining optimal estimate using interval analysis. In the QSE algorithm, we keep the interval in which minimum cost value $\inf\{[c]([p])\}$ in a certain region $[p]$ is smaller than \bar{c} , otherwise discard this interval from the initial region. For those regions including cost values less than \bar{c} , we then bisect the intervals into two parts, and repeat this process until the width of interval is less than ϵ ($\epsilon > 0$). The experimental set-up for QSE is illustrated in Fig. 5(a), with a magnetic intensity map shown in Fig. 5(b). The optimal minimizer is bounded within intervals having minimal sizes given the data and is shown in Fig. 5(c), from which we could see 5 pieces of disjoint $[\hat{p}_i]$ are obtained. The QSE estimation is summarized in Table 1, having \mathbf{D} space as the plain $(x, y) \in \mathbf{D} \subset \mathbb{R}^2$ and $[c]$ space as the set of all cost function intervals $[c]([p])$ from Definition 3.

After evaluating cost functions on the map, more than one local minimizer may be found and bounded into several disjoint intervals $[\hat{p}_i]$ (i.e. as shown in Fig. 5(c)). These intervals form the entire estimate at one algorithmic step, being collected by \hat{S} .

$$\hat{S} = \left[\bigcup_{i=1}^n [\hat{p}_i] \mid \inf\{[c]([\hat{p}_i])\} < \bar{c} \right], \quad n \geq 1 \quad (19)$$

where $[\hat{p}_i]$ is the estimated region that has smallest cost value less than \bar{c} and $[\hat{p}_i] \subseteq [\hat{p}]$. \hat{S} is the union of all estimated intervals having the width vector as $\mathbf{w}(\hat{S})$, from which we can monitor the largest estimate error.

$$\mathbf{w}(\hat{S}) = \sum_{i=1}^n \mathbf{w}([\hat{p}_i]) \quad (20)$$

when n intervals are obtained.

Next, for any single measurement, we show the relationship between the map resolution $(\Delta x, \Delta y)$, derivatives of magnetic

intensity $(\frac{\partial m(x,y)}{\partial x}, \frac{\partial m(x,y)}{\partial y})$, the quality of sensor ($\bar{c} \propto \sigma_{\text{phone}}$), and the estimation accuracy $((k-1)\Delta x, (k-1)\Delta y)$ to evaluate the performance of QSE.

Theorem 1. Let $\Delta x, \Delta y$ be the smallest distances between two nodes along x and y directions respectively on the plane $\mathbf{D} \subset \mathbb{R}^2$. If one estimated area $[\hat{p}]$ contains k nodes on one dimension x and l nodes on y dimension respectively, then the following inequalities hold:

$$\sum_{i=1}^{k-1} \left(\left| \frac{\partial m(x,y)}{\partial x} \right| \right)_i \geq \frac{2(\bar{c} - \inf\{[c]([\hat{p}])\})}{\Delta x} \quad (21a)$$

$$\sum_{i=1}^{l-1} \left(\left| \frac{\partial m(x,y)}{\partial y} \right| \right)_i \geq \frac{2(\bar{c} - \inf\{[c]([\hat{p}])\})}{\Delta y} \quad (21b)$$

where i represents i th node included in $[\hat{p}]$.

Proof. We prove this in one dimension first. Consider a 2-D interval $[p] = ([x_1, x_2], [y_0])^T$, where $[y_0]$ is a fixed interval along y axis, and we have the assumptions that along fixed y direction $y = y_0 \in [y_0]$. Assume $x_1 < x_2$, $c(x_1, y_0) \geq \bar{c}$, $c(x_2, y_0) \geq \bar{c}$. If $[p]$ contains k nodes with the r th node having the smallest cost value $\inf\{[c]([\hat{p}])\}$, then $[p]$ satisfies Eq. (18). Neglecting higher order terms of the Taylor expansion, we have:

$$\begin{aligned} |c(x_{\min}, y_0) - c(x_1, y_0)| &= \left| \sum_{i=1}^{r-1} \left(\frac{\partial c(x,y)}{\partial x} \right)_i \Delta x \right| + \text{H.O.T.} \\ &= \left| \sum_{i=1}^{r-1} \left(\frac{\partial m(x,y)}{\partial x} \right)_i \right| \Delta x \geq \bar{c} - \inf\{[c]([\hat{p}])\} \end{aligned} \quad (22a)$$

$$\begin{aligned} |c(x_2, y_0) - c(x_{\min}, y_0)| &= \left| \sum_{i=r}^{k-1} \left(\frac{\partial c(x,y)}{\partial x} \right)_i \Delta x \right| + \text{H.O.T.} \\ &= \left| \sum_{i=r}^{k-1} \left(\frac{\partial m(x,y)}{\partial x} \right)_i \right| \Delta x \geq \bar{c} - \inf\{[c]([\hat{p}])\}. \end{aligned} \quad (22b)$$

Sum Eqs. (22a) and (22b) above and we have

$$\begin{aligned} \left| \sum_{i=1}^{r-1} \left(\frac{\partial m(x,y)}{\partial x} \right)_i \right| \Delta x + \left| \sum_{i=r}^{k-1} \left(\frac{\partial m(x,y)}{\partial x} \right)_i \right| \Delta x \\ = \sum_{i=1}^{k-1} \left(\left| \frac{\partial m(x,y)}{\partial x} \right| \right)_i \Delta x \geq 2(\bar{c} - \inf\{[c]([\hat{p}])\}). \end{aligned}$$

Hence, expand this into 2-D space with y direction and the similar result is obtained by substituting $\frac{\partial m(x,y)}{\partial x}$ into $\frac{\partial m(x,y)}{\partial y}$ and Theorem 1 is proved. \square

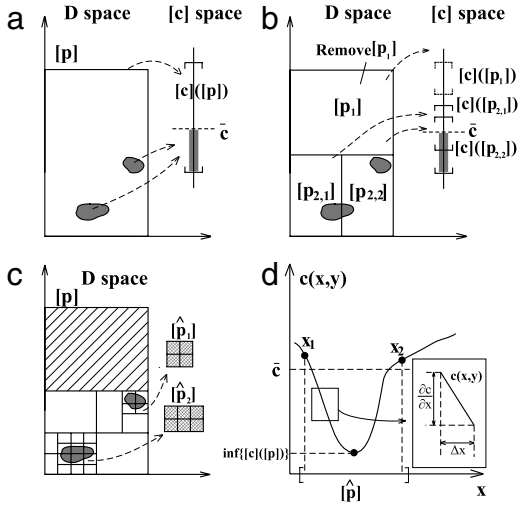


Fig. 4. A schematic representation of QSE estimation algorithm. (a) The initial set up of $[p]$, including the region having the cost value less than \bar{c} ; (b) Bisected regions in $[p]$; (c) The region is processed to the smallest size and the $\hat{[p]}$ is obtained; (d) The illustration to show the proof of Theorem 1.

The proof of Theorem 1 is also illustrated in Fig. 4(d). We use $\bar{c} = 3\sigma_{\text{phone}}$ here as a threshold value to find local minimize intervals. In this case, Eq. (21) gives the relationship between the sharpness of map $\frac{\partial m(x,y)}{\partial x}$, $\frac{\partial m(x,y)}{\partial y}$, quality of sensor σ_{phone} , and the estimate error, which is indicated by the number of nodes k, l . The larger fluctuation local environments have, the sharper the map is, the smaller the final estimation region from the algorithm.

4.2. Dynamic estimation

Some sensor fusion algorithms such as Box Particle Filtering (BPF) (Abdallah, Gning, & Bonnifait, 2008; Gning & Bonnifait, 2008) and Covariance Intersection (CI) (Lazarus et al., 2007) are proposed with respect to the propagation from single interval result. However, as shown in Fig. 5(b), an estimation \hat{S} having multiple disjoint intervals $\hat{[p]}$ contains *true* interval that covers the real position and the *false* intervals that cover the corresponding local minimizers but not the real position, which give redundant intervals and makes it difficult to distinguish the optimal solution. Therefore, the goal of dynamic estimation is to remove those redundant intervals by interacting multiple-time QSE results. Referring to Eq. (20), to reduce the number of parts $\hat{[p]}$ is equivalent to reducing the size of estimation interval $w(\hat{S})$ and hence augment the geolocation.

When collecting multiple measurements while moving in corridors, we need to construct a dynamic path composed of every single geolocation from QSE. As shown in Fig. 3, inputs of dynamic estimation include all QSE results \hat{S}_t from starting point ($t = 0$) to current point ($t = n \cdot \Delta t_s$) by every Δt_s seconds, where $\Delta t_s = 1/f_s$ is the sampling period. Hence, dynamic estimation improves estimation accuracy while keeping the true position inside the region of estimated uncertainty, rather than using only pointwise QSE.

To improve the estimation accuracy, we need to examine each single interval $\hat{[p]}_i \subset \hat{S}$ and to remove those redundant ones by monitoring their propagation. Generally, the true position will follow the route on the map and also the true interval. Hence, true intervals must have the property of continuity when multiple estimates are obtained, under the assumption that the smallest interval estimate width $w([p])$ is much larger than the true displacement vector between two neighboring points Δd from successive measurements.

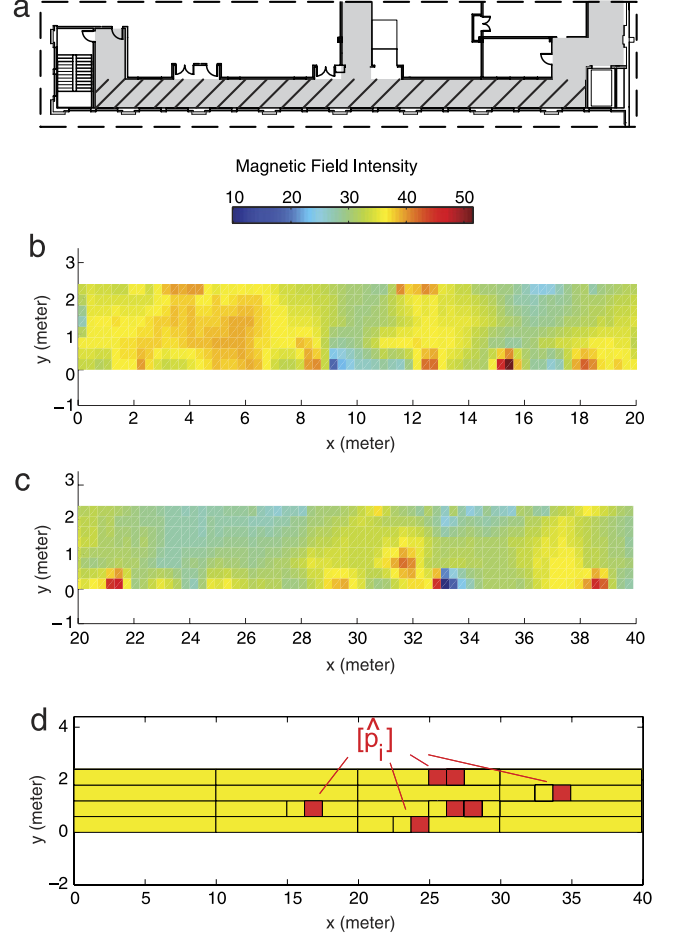


Fig. 5. Illustration of QSE. (a) The experiment region is the shaded area on the floor plan. (b) The magnetic intensity map after collecting data on each node on the grid. (c) The yellow region represents the corridor and the red region represents the set of all static estimated intervals \hat{S} , including each single interval as a local minimum region $\hat{[p]}_i$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Theorem 2. Suppose there are two neighboring grid points on the plane (x_k, y_k) and (x_{k+1}, y_{k+1}) , and the displacement vector between them is:

$$\Delta d = (\Delta d_x, \Delta d_y)^T = (x_{k+1} - x_k, y_{k+1} - y_k)^T.$$

If $\Delta d_x > 0$, $\Delta d_y > 0$ and $\Delta d_x < w_x([p])$, $\Delta d_y < w_y([p])$, then for $(x_k, y_k) \in [p_k]$ and $(x_{k+1}, y_{k+1}) \in [p_{k+1}]$, $[p_k] \cap [p_{k+1}] \neq \emptyset$.

The proof of Theorem 2 is shown in Appendix B.

Corollary 1. Consider any interval set in two neighboring solution sets from the QSE result: \hat{S}_t estimated from the measurement z_m^t at t and \hat{S}_{t+1} estimated from the next measurement z_m^{t+1} at $t + 1$. Suppose $\hat{[p]}_{t,i} \subset \hat{S}_t$ and $\hat{[p]}_{t+1,j} \subset \hat{S}_{t+1}$, with the assumption that $(x_k, y_k) \in \hat{S}_t$ and $(x_{k+1}, y_{k+1}) \in \hat{S}_{t+1}$. If $\hat{[p]}_{t+1,j} \cap \hat{S}_t = \emptyset$, then $(x_{k+1}, y_{k+1}) \notin \hat{[p]}_{t+1,j}$ holds.

Proof. We prove this by contradiction. Suppose we know that $(x_k, y_k) \in \hat{S}_k$, and if $(x_{k+1}, y_{k+1}) \in \hat{[p]}_{k+1,j}$. Hence, from Theorem 1, we have $\hat{[p]}_{k+1,j} \cap \hat{[p]}_{k,i} \neq \emptyset$. Since $\hat{[p]}_{k+1,j} \subset \hat{S}_{k+1} \Rightarrow \hat{[p]}_{k+1,j} \cap \hat{S}_k \neq \emptyset$, and this is in contradiction with the assumption. \square

Therefore, we need to track each $\hat{[p]}_i$ and to see if every part in the next estimation have any intersections with the current interval set, and remove the one that has nothing related to the former estimation. Corrected intervals are put back into \hat{S} and all

Table 2

Algorithm of dynamic estimation.

Algorithm 2: dynamic estimation
1. for $i = 1 : n$
2. for $k = 1 : q$
3. if $[\hat{\mathbf{p}}_{i,k}] \cap [\bigcup_{j=1}^m [\hat{\mathbf{p}}_{i-1,j}]] \neq \emptyset$
4. $[\tilde{\mathbf{p}}_{i,l}] = [\hat{\mathbf{p}}_{i,k}], l++$
6. end if
7. end for
8. $\hat{\mathbf{S}}_i \leftarrow ([\tilde{\mathbf{p}}_{i,l}])^T, l = 1, \dots, r$
9. end for
10. $\hat{\mathbf{C}} = \bigcup_{i=1}^n \hat{\mathbf{S}}_i$

updated estimations are collected by $\hat{\mathbf{C}}$. Without loss of generality, we choose a pair of two-step QSE, $[\hat{\mathbf{p}}_{i-1,j}] \in \hat{\mathbf{S}}_{i-1}$ and $[\hat{\mathbf{p}}_{i,k}] \in \hat{\mathbf{S}}_i$ to present the algorithm in Table 2.

4.3. Result of dynamic estimation

We carry out the experiment based on our floor plan in the mechanical engineering building on Purdue campus, having path as the shaded area shown in Fig. 2(a). In the practical estimation process we use parameters as $\bar{c} = 1.0\mu T > 3 * \sigma_{phone}$ for the upper bound of the cost function, and $\epsilon = [1.10, 0.60]m$ to bound the size of wrappers on two directions. Due to the physical meaning of σ_{phone} , a value $\bar{c} \geq 3 * \sigma_{phone}$ is suggested. If \bar{c} is extremely large, the estimation process becomes meaningless; if it is very small, then the uncertainty intervals cannot be intersected in dynamic estimation. Hence, we need some imprecision on all individual sensors. To finish the calculation and illustration of interval analysis, parts of the algorithms are realized by using *Intlab* (Hargreaves, 2002; Rump, 1999) embedded in Matlab. The dynamic estimation result is shown in Fig. 6. As introduced in Theorem 1, those regions having relatively large sharpness of magnetic intensity tend to narrow the estimation size is shown in the experiment result, which means to utilize local fluctuations gives us effectiveness to improve the indoor geolocation.

5. Sensor fusion with IMU

As described in the last section, the reduce of the estimate error becomes more efficient after sufficient readings from magnetometers, especially the evidence of passing a strong magnetic field area. On the contrast, estimating position in GPS-denied area with IMU independently without any auxiliary methods such as ZUPT (Bebek et al., 2010) or human gait model (Cui & Ariyur, 2011; Jirawimut et al., 2003) will propagate the error over a long time, cube of travel duration times with error increasing as the measurement bias. This condition is more common since it is not easy to measure or estimate the body locomotion or to find foot velocity when cell phone users hold their devices in hand. Hence, since the dynamic estimation algorithm offers the estimated region after every measurement from magnetometer, while the IMU has the advantage of introducing precise estimation in a short time period, we fuse the magnetometer with the IMU to improve the accuracy of estimation, with a sampling rate of $\Delta t = 0.06$ s from both magnetometer and IMU. The fusion algorithm includes two phases, the inertial update (IU) and the magnetometer update (MU). The structure of the algorithm is presented in Fig. 7, and we integrate the two as separate interval estimates.

5.1. Inertial update (IU)

We use the embedded IMU from a smart phone that can measure acceleration and angular rate both in three directions, from

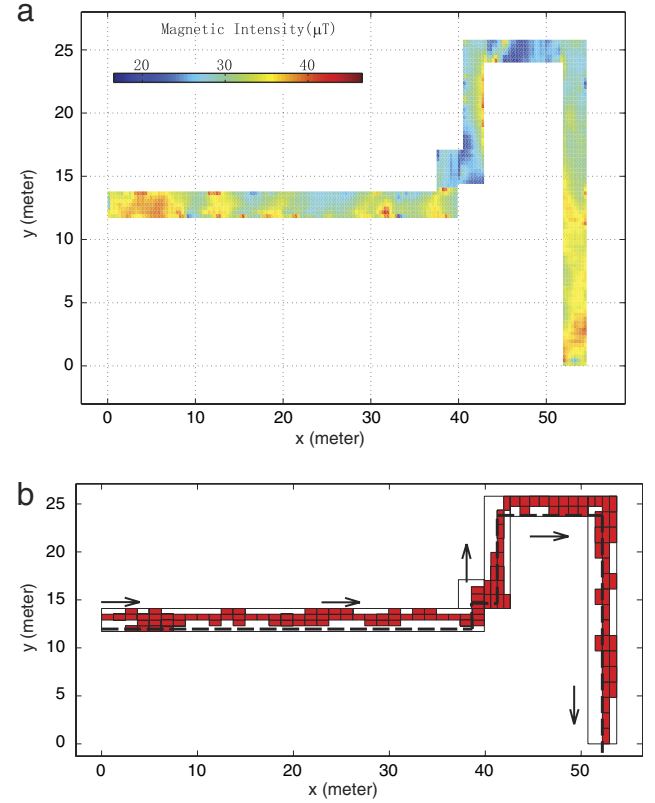


Fig. 6. Result of dynamic estimation in mechanical engineering corridor. (a) The magnetic intensity map of the corridor; (b) The red region represents the estimated interval from dynamic estimation. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

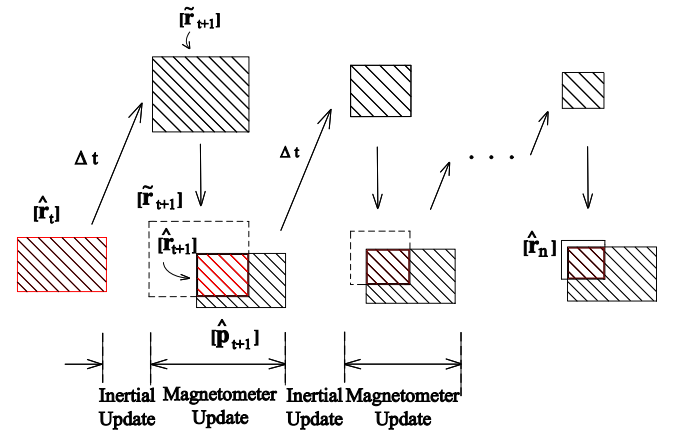


Fig. 7. The structure of inertial update and magnetometer update. The region wrapped by dashed line represents the box of the inertial update result, and the red-shaded region represents the result after magnetometer update processed by intersecting the current estimation from dynamic estimation. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

which vectors of acceleration \mathbf{a}^i and angular rate \mathbf{w}^i are collected as:

$$\mathbf{a}^i = (a_x^i, a_y^i, a_z^i)^T; \quad \mathbf{w}^i = (w_x^i, w_y^i, w_z^i)^T \quad (23)$$

where the superscript i represents the inertial coordinate (body coordinate) used to distinguish from the navigation coordinate having superscript n . The transformation from inertial coordinate O_i to

the navigation coordinate O_n is the following equation. $R_{3 \times 3}$ states are estimated via Kalman Filter with some pre-assumptions, and the structure to estimate is attached in [Appendix A](#).

$$\mathbf{a}^n = R_{3 \times 3} \mathbf{a}^i \quad (24)$$

where $\mathbf{a}^n = (a_x^n, a_y^n, a_z^n)^T$ and $\mathbf{a}^i = (a_x^i, a_y^i, a_z^i)^T$. Recall that we note $[\hat{\mathbf{p}}_t] \subset \hat{\mathbf{S}}_t$ as estimate intervals after dynamic estimation process. Similarly, we use another notation $\hat{\mathbf{R}}_t$ as estimate interval in sensor fusion process, with the initial condition defined as $\hat{\mathbf{R}}_0 = \hat{\mathbf{S}}_0$ which means that $\hat{\mathbf{R}}_0$ is not intersected with other intervals yet and hence holds the same estimate as $\hat{\mathbf{S}}_0$. Note that after dynamic estimation process, one or more $[\hat{\mathbf{p}}_t]$ are obtained in $\hat{\mathbf{S}}_t$. Similarly, we could assume that at a certain time step t , one or more intervals are obtained in $\hat{\mathbf{R}}_t$, and one can choose one sub-region in $\hat{\mathbf{R}}_t$, noted as $[\hat{\mathbf{r}}_t] \subset \hat{\mathbf{R}}_t$.

Assume the updated estimate set as $\tilde{\mathbf{R}}_{t+1}$, and put $[\hat{\mathbf{r}}_t] \subset \hat{\mathbf{R}}_t$ and $[\tilde{\mathbf{r}}_{t+1}] \subset \tilde{\mathbf{R}}_{t+1}$ into dynamic model, we have:

$$\forall \hat{\mathbf{x}}_t^n = (\hat{x}_t^n, \hat{y}_t^n)^T \in [\hat{\mathbf{r}}_t] \subset \hat{\mathbf{R}}_t,$$

$$\forall \tilde{\mathbf{x}}_{t+1}^n = (\tilde{x}_{t+1}^n, \tilde{y}_{t+1}^n)^T \in [\tilde{\mathbf{r}}_{t+1}] \subset \tilde{\mathbf{R}}_{t+1}$$

$$\tilde{\mathbf{x}}_{t+1}^n = \hat{\mathbf{x}}_t^n + \mathbf{v}_t^n \cdot \Delta t + \frac{1}{2}(\mathbf{a}_t^n + \mu_a^n) \cdot \Delta t^2 \quad (25)$$

$$\mathbf{v}_t^n = \mathbf{v}_{t-1}^n + (\mathbf{a}_{t-1}^n + \mu_a^n) \Delta t \quad (26)$$

with an initial condition of $v_{x,0}^n = v_{y,0}^n = 0$, and μ_{ax}^n, μ_{ay}^n represent the measurement error in accelerometer. $\mu_a^n = (\mu_{ax}^n, \mu_{ay}^n)^T \sim (\mathbf{0}, v_a^2)$. To analyze system dynamics model via interval analysis, we re-write Eq. (25) above into intervals, and we have:

$$[\hat{\mathbf{r}}_t] \subset \hat{\mathbf{R}}_t \quad \text{and} \quad [\tilde{\mathbf{r}}_{t+1}] \subset \tilde{\mathbf{R}}_{t+1}$$

$$[\tilde{\mathbf{r}}_{t+1}] = [\hat{\mathbf{r}}_t] + \frac{1}{2}[\mathbf{a}] \Delta t^2 + [\mathbf{v}] \Delta t \quad (27)$$

where $[\mathbf{a}] = [\mathbf{a}_t^n - v_a, \mathbf{a}_t^n + v_a]$ and v_a is the error bound vector from the accelerometer. In Eq. (27), The basic operations of intervals here follow Eq. (8) introduced in Section 3. We use $[\tilde{\mathbf{r}}_{t+1}]$ to represent one interval after inertial update, and use $\tilde{\mathbf{R}}_{t+1}$ to collect all $[\tilde{\mathbf{r}}_{t+1}]$. Hence we have:

$$\tilde{\mathbf{R}}_{t+1} = \bigcup_{i=1}^n [\tilde{\mathbf{r}}_{t+1,i}] \quad (28)$$

if n pieces of $[\hat{\mathbf{r}}_{t+1}]$ are obtained. Note that the number of segments $[\tilde{\mathbf{r}}_{t+1}] \subset \tilde{\mathbf{R}}_{t+1}$ is the same as $[\hat{\mathbf{r}}_t] \subset \hat{\mathbf{R}}_t$, so the number of intervals from estimation is not decreased in this phase. Eq. (27)–(28) with initial condition form the phase of IU, in which results from dynamic estimation are propagated with readings from inertial measurement and obtained through $\tilde{\mathbf{R}}_{t+1}$.

5.2. Magnetometer update (MU)

The inertial update gives the propagated interval $\tilde{\mathbf{R}}_{t+1}$ after every Δt seconds, while a new estimate processed by dynamic estimation algorithm using magnetometer data at time $t + \Delta t$ has been obtained as $\hat{\mathbf{S}}_{t+1}$. We then intersect these two interval sets to obtain the updated estimation, noted as $\hat{\mathbf{R}}_{t+1}$, and use this interval as the updated estimation from MU. Hence, we have:

$$\hat{\mathbf{R}}_{t+1} = \tilde{\mathbf{R}}_{t+1} \cap \hat{\mathbf{S}}_{t+1}. \quad (29)$$

Remark 3. As from Eq. (29), the size of new estimation $\hat{\mathbf{R}}_{t+1}$ has the width vector $\mathbf{w}(\hat{\mathbf{R}}_{t+1}) = \mathbf{w}(\tilde{\mathbf{R}}_{t+1} \cap \hat{\mathbf{S}}_{t+1}) \leq \mathbf{w}(\hat{\mathbf{S}}_{t+1})$.

This computation is independent of the time vector and the IU so we note this phase as MU. Eq. (29) forms the process of MU

which gives an entire update of one fusion cycle and the $\hat{\mathbf{R}}_{t+1}$ as the updated estimate after one cycle of both IU and MU, which can be propagated in Eq. (27) for following cycles.

Having obtained $\tilde{\mathbf{R}}_{t+1}$ from inertial update, the relationship between $\tilde{\mathbf{R}}_{t+1}$ and the dynamic update at $t + 1$ time step $\hat{\mathbf{S}}_{t+1}$ need to be identified first to find further operation and fuse. To better understand the displacement between them, recall [Theorem 2](#), result from the dynamic estimation must have the property that $\hat{\mathbf{S}}_t \cap \hat{\mathbf{S}}_{t+1} \neq \emptyset$. One can assume that $\tilde{\mathbf{R}}_t = \hat{\mathbf{S}}_t$ and choose $[\hat{\mathbf{p}}_t] \subset \hat{\mathbf{S}}_t$ and $[\hat{\mathbf{p}}_{t+1}] \subset \hat{\mathbf{S}}_{t+1}$, and note the intersection as:

$$[\hat{\mathbf{q}}_{t,t+1}] = [\hat{\mathbf{p}}_t] \cap [\hat{\mathbf{p}}_{t+1}]. \quad (30)$$

Since both $[\hat{\mathbf{p}}_t]$ and $[\hat{\mathbf{p}}_{t+1}]$ are boxes, $[\hat{\mathbf{q}}_{t,t+1}]$ is also a box having boundaries aligned with coordinate axis, and is written as $[\mathbf{q}]$ for convenience. The width vector of $[\mathbf{q}]$ is:

$$\mathbf{w}([\mathbf{q}]) = (w_x([\mathbf{q}]), w_y([\mathbf{q}]))^T$$

$$w_x([\mathbf{q}]) \leq w_x([\hat{\mathbf{p}}_t]) \quad \text{and} \quad w_y([\mathbf{q}]) \leq w_y([\hat{\mathbf{p}}_t]). \quad (31)$$

Moreover, from Eq. (27), we want to isolate the displacement interval and re-write Eq. (26) into:

$$[\tilde{\mathbf{r}}_{t+1}] = [\hat{\mathbf{r}}_t] + [\Delta \mathbf{D}]$$

$$[\Delta \mathbf{D}] = \frac{1}{2}[\mathbf{a}] \Delta t^2 + [\mathbf{v}] \Delta t$$

$$= \left[\frac{1}{2}(\mathbf{a}_t^n - v_a) \Delta t^2, \frac{1}{2}(\mathbf{a}_t^n + v_a) \Delta t^2 \right] + [\mathbf{v}] \Delta t. \quad (32)$$

Hence, $\inf([\Delta \mathbf{D}])$ and $\sup([\Delta \mathbf{D}])$ represent the shortest and longest displacements estimated from inertial measurement, and we can analyze $\Delta \mathbf{D}$ to justify if $[\tilde{\mathbf{r}}_{t+1}]$ can be used to interact with $[\hat{\mathbf{p}}_t] \subset \hat{\mathbf{S}}_{t+1}$ and evaluate the performance of possible results. Use x axis as an example, if the smallest displacement is larger than the length $w_x([\hat{\mathbf{p}}_t]) + w_x([\hat{\mathbf{p}}_{t+1}]) - w_x([\mathbf{q}])$, the interval $[\tilde{\mathbf{r}}_{t+1}]$ has no intersection with $[\hat{\mathbf{p}}_{t+1}]$ and hence $[\tilde{\mathbf{r}}_{t+1}] \cap [\hat{\mathbf{p}}_{t+1}] = \emptyset$.

On the opposite moving direction, this inequality becomes $\inf([\Delta D_x]) \geq w_x([\mathbf{q}])$. Since we know that $w_x([\mathbf{q}]) \leq w_x([\hat{\mathbf{p}}_t])$ and $w_x([\mathbf{q}]) \leq w_x([\hat{\mathbf{p}}_{t+1}])$, then the conclusion is drawn by:

$$\inf([\Delta D_x]) \geq \max(w_x([\hat{\mathbf{p}}_t]) + w_x([\hat{\mathbf{p}}_{t+1}]) - w_x([\mathbf{q}]), w_x([\mathbf{q}]))$$

$$\geq w_x([\hat{\mathbf{p}}_t]) + w_x([\hat{\mathbf{p}}_{t+1}]) - w_x([\mathbf{q}]). \quad (33)$$

From the description above, we can draw a new theorem to predict the estimation accuracy of $[\hat{\mathbf{r}}_{t+1}]$ based on the sensor fusion algorithm.

Theorem 3. Along the k direction ($k = x, y$),

Case 1. If $\inf([\Delta D_k]) \geq w_k([\hat{\mathbf{p}}_t]) + w_k([\hat{\mathbf{p}}_{t+1}]) - w_k([\mathbf{q}])$, then

$$[\tilde{\mathbf{r}}_{t+1}] \cap [\hat{\mathbf{p}}_{t+1}] = \emptyset, \quad \mathbf{w}([\tilde{\mathbf{r}}_{t+1}]) = \mathbf{0}.$$

Case 2. If $\begin{cases} \inf([\Delta D_k]) \leq \inf([\mathbf{q}]) - \inf(\hat{\mathbf{p}}_{t,k}) \\ \sup([\Delta D_k]) \geq \sup([\mathbf{q}]) - \sup(\hat{\mathbf{p}}_{t,k}) \end{cases}$ then,

$$w_k([\hat{\mathbf{r}}_{t+1}]) \geq w_k([\mathbf{Q}]).$$

The proof of [Theorem 3](#) is given in [Appendix C](#). [Theorem 3](#) gives conditions when the updated results are unavailable or trivial. In case 1, the intersection $[\hat{\mathbf{r}}_{t+1}] = [\tilde{\mathbf{r}}_{t+1}] \cap [\hat{\mathbf{p}}_{t+1}]$ is empty, and we cannot use the inertial estimate. Practically, however, this never happens if we bound the estimate intervals around 1 m and $\Delta t = 0.06$ s, we did not find any large acceleration such that Eq. (33) holds. In the second case, it happens if $[\Delta \mathbf{D}]$ has a large range. Recall Eq. (32) and Case 2, if $[\mathbf{v}]$ is well-bounded, then if the error in accelerometer is too large then Eq. (33) must be unsatisfied but Case 2 is satisfied. The result $[\hat{\mathbf{r}}_{t+1}]$ exists, having widths even larger than $\mathbf{w}([\mathbf{Q}])$, which means that the result interval is not improved

by this algorithm. Hence, **Theorem 3** gives the effective operation range of the algorithms fusing with IMU and magnetometer.

Since the estimation is obtained by bisecting the rough estimated region and bounded via Eq. (19), there exists the possibility that $[\hat{\mathbf{p}}_t] = [\hat{\mathbf{p}}_{t+1}]$ from two dynamic estimation intervals. In this case, we simplify **Theorem 3** and obtain **Corollary 2**.

Corollary 2. If $[\hat{\mathbf{p}}_t] = [\hat{\mathbf{p}}_{t+1}]$, then **Theorem 3** becomes: Along the k direction ($k = x, y$),

Case 1. If $\inf([\Delta D_k]) \geq w_k([\hat{\mathbf{p}}_t])$ then $[\hat{\mathbf{r}}_{t+1}] \cap [\hat{\mathbf{p}}_{t+1}] = \emptyset$.

Case 2. If $\inf([\Delta D_k]) \leq 0$ and $\sup([\Delta D_k]) \geq 0$, then $w_k([\hat{\mathbf{r}}_{t+1}]) \geq w_k([\mathbf{q}])$.

This covers the two cases in **Theorem 3** as well. An important issue here is that Case 1 in **Corollary 2** is theoretically possible when a sudden displacement occurs, but actually in contract with the assumption in **Theorem 2**. In our practical work, the Case 1 has not been satisfied but may be satisfied further when moving fast or being lack of sampling rate, which is out of discussion in this paper.

5.3. Comparison with particle filter

Particle filter is a well-developed algorithm over past years as an effective method using Monte Carlo Localization (MCL) technique to deal with non-linear tracking problem (Doucet & Johansen, 2008; Gustafsson et al., 2002; Ristic, Arulampalam, & Gordon, 2004), and has been applied to the magnetic field based localization (Haverinen & Kempainen, 2009). Here we modify the model to fuse the IMU data and compare the result with the interval filter as presented above. Based on Eq. (24), if $\mathbf{x}_t^n = (x_t^n, y_t^n, z_t^n)^T$ represents the states at time t , and the dynamic model can be taken as:

$$\mathbf{x}_{t+1}^n = \mathbf{x}_t^n + \frac{1}{2} \mathbf{a}_t^n \cdot \Delta t^2 + \mathbf{v}_t^n \cdot \Delta t + \mathbf{v}^n \quad (34)$$

where $\mathbf{v}^n = (v_x^n, v_y^n, v_z^n)^T$ represents the system error on navigation frame. We assume independent zero-mean white noise on inertial frame $\mathbf{v}^i = (v_x^i, v_y^i, v_z^i)^T$ having the property of uncorrelated variation: $\mathbf{v}^i = \text{diag}(\sigma_x^2, \sigma_y^2, \sigma_z^2)$. Then similar to Eq. (24), system noise is transformed from inertial frame to the navigation frame by $\mathbf{v}^n = \mathbf{R}_{3 \times 3} \mathbf{v}^i$. The measurement function is modeled with neglecting the impact of z direction:

$$z_m = m(x_t^n, y_t^n) + \omega \quad (35)$$

where the ω stands for the measurement noise and assumed as zero-mean white noise, having $\text{var}(\omega) = \sigma_r^2$.

Particle filter starts with a set of pre-estimated positions and update for the next step, considering the IMU data. The measurement function (35) will be updated then to introduce a set of normally distributed measurement. The real measurement z_m will be read into the process and the likelihood of the measurement given updated state, $p(z_m | (x_t^n, y_t^n))$ is obtained by:

$$p(z_m | (x_t^n, y_t^n)) = \frac{1}{\sqrt{2\pi\sigma_r^2}} \exp\left(-\frac{(z_m - m(x_t^n, y_t^n))^2}{2\sigma_r^2}\right).$$

This updates the weight of the estimated measurement with respect to the real measurement. After normalizing this probability, we use the *systematic resampling* method to resample to obtain N new equally-weighted particles to prevent degeneracy. See Chapter 3 of Doucet and Johansen (2008) for a comprehensive derivation of the resampling.

5.4. Sensor fusion results

We carry out sensor fusion using dynamic estimation results, as shown in Fig. 5, and collect both the acceleration and angular rate

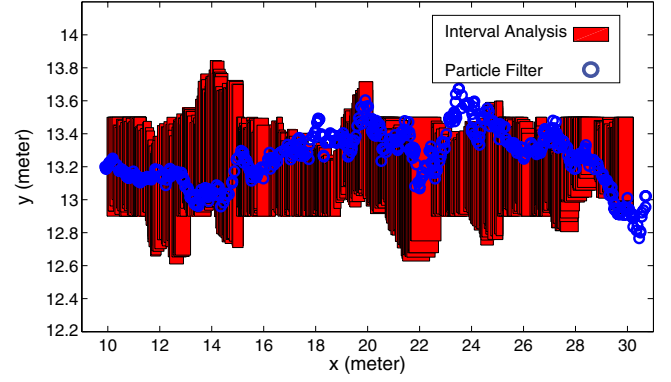


Fig. 8. Sensor fusion result: The red boxes represent the estimated intervals from the filter of interval analysis, having some small sized ones covered by the larger ones. The blue line represents the estimated mean value position from particle filter. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

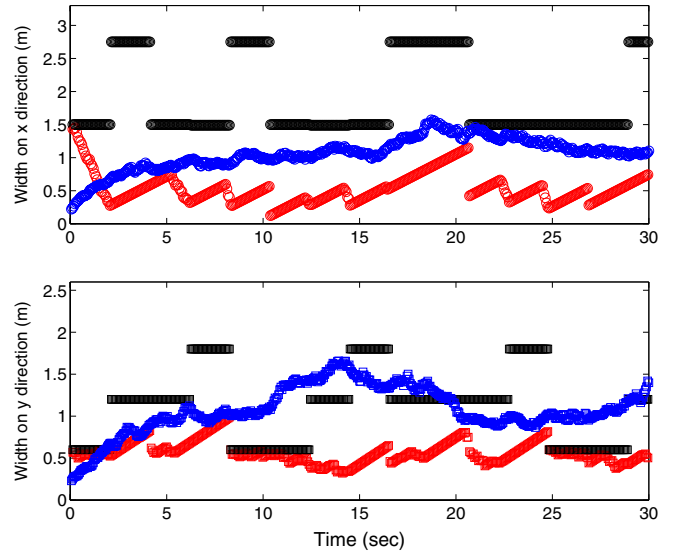


Fig. 9. Sizes of sensor fusion result: The upper sub-plot shows the width on X direction and the bottom sub-plot shows the width on Y direction. On each of the plot, the black line represents the width of estimated interval from dynamic estimation, the red line represents the width of fusion via interval analysis filter, and the blue line represents the fusion from particle filter. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

data simultaneously with a sampling rate of $f_s = 1/\Delta t = 16$ Hz. To show the improvement of result clearly, we use a piece of the path from the fusion process. After running IU and MU phases, estimate intervals are obtained and shown in Fig. 8. Moreover, the particle filtering algorithm using initial estimated variance $\sigma_{x0} = 0.2$ m, $\sigma_{y0} = 0.2$ m and the variance of process $\sigma_x = \sigma_y = 1$ m, $\sigma_r = 0.268 \mu\text{T}$ is also implemented, with $N = 1000$ particles generated. The mean value of each estimate from particle filter is shown as blue circle in Fig. 8, and the variance on both directions are shown in Fig. 9, from which the estimate error from interval analysis are smaller than dynamic estimation result and is even smaller than the particle filter except the initial steps. The smallest estimate error on x, y direction is 0.25 m and 0.20 m respectively, which sharply reduce the estimation size while cell phone user is moving.

6. Conclusion and future work

In this paper, we propose new methods to bound the optimal estimation by using the magnetic intensity map. These methods

are based on the framework of interval analysis which is good for map matching problems. For a moving cell phone user, the dynamic estimation judges results from QSE, removes redundant intervals, and propagates the rest to generate a path estimate, from which the estimate error is effectively reduced. Besides, we develop a method to fuse magnetometer measurements with IMU to improve the geolocation, and the experiment data shows the effectiveness of this method compared to particle filter. As part of future work, we intend to build up the map autonomously with help of stride detection and ultimately realize SLAM based on real-time map update, as well as apply the fusion algorithm to other map-based estimation problems and study the computational optimization of the algorithms.

Acknowledgments

We thank Andrew Kirmse and Madhukar Korupolu for detailed discussions that helped us formulate the crux of this problem.

Appendix A. Rotation matrix and KF loop

Use ψ , θ , and ϕ to represent rotation angles along x , y and z axes in inertial frame, and the 3×3 rotation matrix $R_{3 \times 3}$ can be presented as:

$$R_{3 \times 3} = \begin{pmatrix} C\theta C\phi & S\psi S\theta C\phi - C\psi S\phi & C\psi S\theta C\phi + S\psi S\phi \\ C\theta S\phi & S\psi S\theta S\phi + C\psi C\phi & C\psi S\theta S\phi - S\psi C\phi \\ -S\theta & S\psi C\theta & C\psi C\theta \end{pmatrix}$$

with the abbreviation of $C\psi = \cos(\psi_t)$, $S\psi = \sin(\psi_t)$ and the same for θ and ϕ . Three states are estimated via Kalman Filter loops, starting with settings as: $\mathbf{x}_k = (\psi_k, \omega_{x,k}, \theta_k, \omega_{y,k}, \phi_k, \omega_{z,k})^T$, while the measurement vector is noted as \mathbf{z}_k . Thus, the dynamic equation and the measurement equation is modeled as:

$$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \omega_k \quad (A.1)$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \nu_k \quad (A.2)$$

$\Phi_k = \text{diag}(\Phi_x, \Phi_y, \Phi_z)$, $\Phi_x = \Phi_y = \Phi_z = \begin{pmatrix} 1 & 0.02 \\ 0 & 1 \end{pmatrix}$, and $\mathbf{H}_k = [0 \ 1 \ 0 \ 1 \ 0 \ 1]$. The attributes of system noise and measurement noise are:

$$\mathbf{Q}_k = \text{diag}(\mathbf{Q}_{x,k}, \mathbf{Q}_{y,k}, \mathbf{Q}_{z,k}), \quad \mathbf{R}_k = \text{diag}(\mathbf{R}_{x,k}, \mathbf{R}_{y,k}, \mathbf{R}_{z,k})$$

$$\mathbf{Q}_{x,k} = \mathbf{Q}_{y,k} = \begin{pmatrix} 0.001 & 0 \\ 0 & 0.1 \end{pmatrix}, \quad \mathbf{Q}_{z,k} = \begin{pmatrix} 0.004 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\mathbf{R}_{x,k} = \mathbf{R}_{y,k} = \begin{pmatrix} 0.001 & 0 \\ 0 & 0.5 \end{pmatrix}, \quad \mathbf{R}_{z,k} = \begin{pmatrix} 0.004 & 0 \\ 0 & 4 \end{pmatrix}.$$

See Brown and Hwang (1997) about notations, definitions, and derivations in a standard KF loop, which starts with computing Kalman gain:

$$\mathbf{K}_k = \mathbf{P}_k^{-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}. \quad (A.3)$$

Then update estimate with measurement \mathbf{z}_k and compute the error covariance for updated estimate:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \quad (A.4)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-. \quad (A.5)$$

Finally project to the next loop:

$$\hat{\mathbf{x}}_{k+1} = \Phi_k \hat{\mathbf{x}}_k \quad (A.6)$$

$$\mathbf{P}_{k+1}^- = \Phi_k \mathbf{P}_k \Phi_k^T + \mathbf{Q}_k \quad (A.7)$$

and these are proceeded to the next loop from Eq. (A.7).

Appendix B. Proof of Theorem 2

Proof. We prove this on one dimension and the other holds too. Along x direction, $x_{k+1} = x_k + \Delta d_x$, set an interval set for the distance $[\Delta d_x] = [\inf\{\Delta d_x\}, \sup\{\Delta d_x\}]$ and examine the term x_c :

$$\begin{aligned} x_c &= \inf\{\mathbf{x}_{k+1}\} - \sup\{\mathbf{x}_k\} = \inf\{\mathbf{x}_k + \Delta d_x\} - \sup\{\mathbf{x}_k\} \\ &= (\sup\{\mathbf{x}_k\} - w(\mathbf{x})) + \inf\{\Delta d_x\} - \sup\{\mathbf{x}_k\} \\ &\leq \sup\{\Delta d\} - w(\mathbf{x}) < 0. \end{aligned}$$

And same as on y direction: $y_c = \inf\{\mathbf{y}_k\} \cap \{\mathbf{y}_{k+1}\} < 0 \Rightarrow [\mathbf{p}_k] \cap [\mathbf{p}_{k+1}] \neq \emptyset$. \square

Appendix C. Proof of Theorem 3

Proof. Choose x direction as an example, and proof of case 1 is given in Eq. (34), and since $[\tilde{\mathbf{r}}_{t+1}] \cap [\hat{\mathbf{p}}_{t+1}] = \emptyset$, the intersection has width of zero, $w([\tilde{\mathbf{r}}_{t+1}]) = 0$.

For case 2:

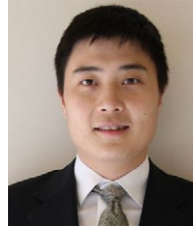
$$\begin{aligned} \inf(\Delta D_x) &\leq \inf([Q_x]) - \inf([\hat{p}_{t,x}]) \\ \Rightarrow \inf(\tilde{r}_{t+1,x}) &= \inf([\hat{p}_{t,x}]) + \inf(\Delta D_x) \leq \inf([Q_x]) \\ \sup(\Delta D_x) &\leq \sup([Q_x]) - \sup([\hat{p}_{t,x}]) \\ \Rightarrow \sup(\tilde{r}_{t+1,x}) &= \sup([\hat{p}_{t,x}]) + \sup(\Delta D_x) \geq \sup([Q_x]). \end{aligned}$$

From two equations above, $[Q_x] \subset [\tilde{r}_{t+1,x}]$. Recall Eq. (30) and hence $[Q_x] \subset [\hat{r}_{t+1,x}] \Rightarrow w_x([\hat{\mathbf{r}}_{t+1}]) \geq w_x([\mathbf{Q}])$. Similarly, replace x into y direction and the two cases above have the same conclusion. \square

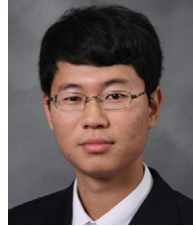
References

- Abdallah, F., Gning, A., & Bonnifait, P. (2008). Box particle filtering for nonlinear state estimation using interval analysis. *Automatica*, 44(3), 807–815.
- Arulampalam, M. S., et al. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2), 174–188.
- Bajaj, R., Ranaweera, S. L., & Agrawal, D. P. (2002). GPS: location-tracking technology. *Computer*, 4, 92–94.
- Bebek, Ö., et al. (2010). Personal navigation via high-resolution gait-corrected inertial measurement units. *IEEE Transactions on Instrumentation and Measurement*, 59(11), 3018–3027.
- Bekkali, A., Sanson, H., & Matsumoto, M. (2007). RFID indoor positioning based on probabilistic RFID map and Kalman filtering. In *Proceedings of IEEE international conference on wireless and mobile computing, networking and communications* (pp. 21–27).
- Bonnet, S., & Hélio, R. (2007). A magnetometer-based approach for studying human movements. *IEEE Transactions on Biomedical Engineering*, 54(7), 1353–1355.
- Brown, R. G., & Hwang, P. Y. C. (1997). *Introduction to random signals and applied Kalman filtering* (3rd ed.). John Wiley & Sons.
- Bshara, M., Orguner, U., Gustafsson, F., & Van Biesen, L. (2010). Fingerprinting localization in wireless networks based on received-signal-strength measurements: a case study on WiMAX networks. *IEEE Transactions on Vehicular Technology*, 59(1), 283–294.
- Burnett, J., & Yaping, P. D. (2002). Mitigation of extremely low frequency magnetic fields from electrical installations in high-rise buildings. *Building and Environment*, 37, 769–775.
- Byun, S. H., Hajj, G. A., & Young, L. E. (2001). Assessment of GPS signal multipath interference. In *Proceedings of the 2002 national technical meeting of the institute of navigation* (pp. 694–705).
- Chung, J. et al. (2011). Indoor location sensing using geo-magnetism. In *Proceedings of the 9th international conference on mobile systems, applications, and services*. Bethesda, Maryland, USA (pp. 141–154).
- Cui, Y., & Ariyur, K. B. (2011). Bounding inertial drift with human gait dynamics for personal navigation. In *Proceedings of IEEE international system conference*. Montreal, Canada (pp. 28–33).
- Doucet, A., & Johansen, A. M. (2008). A tutorial on particle filtering and smoothing: fifteen years later.
- Fang, L., et al. (2005). Design of a wireless assisted pedestrian dead reckoning system—the NavMote experience. *IEEE Transactions on Instrumentation and Measurement*, 54(6), 2342–2358.
- Fang, S.-H., Lin, T.-N., & Lee, K.-C. (2008). A novel algorithm for multipath fingerprinting in indoor WLAN environments. *IEEE Transactions on Wireless Communications*, 7(9), 3579–3588.
- Finlay, C. C., et al. (2010). International geomagnetic reference field: the eleventh generation. *Geophysical Journal International*, 183, 1216–1230.

- Gning, A., Abdallah, F., & Bonnifait, P. (2007). A new estimation method for multisensor fusion by using interval analysis and particle filtering. In *Proceedings of IEEE international conference on robotics and automation*. Italy (pp. 3844–3849).
- Gning, A., & Bonnifait, Ph. (2008). Constraints propagation techniques on intervals for a guaranteed localization using redundant data. *Automatica*, 44(3), 807–815.
- Gustafsson, F., & Gunnarsson, F. (2005). Mobile positioning using wireless networks. *IEEE Signal Processing Magazine*, 22(4), 41–53.
- Gustafsson, F., Nordlund, P.-J., et al. (2002). Particle filters for positioning, navigation and tracking. *IEEE Transactions on Signal Processing*, 50(2), 425–437.
- Hargreaves, G. I. (2002). *Interval analysis in Matlab*. (thesis), Manchester, England: University of Manchester.
- Haverinen, J., & Kemppainen, A. (2009). Global indoor self-localization based on the ambient magnetic field. *Robotics and Autonomous Systems*, 57(10), 1028–1035.
- Horn, J. P., et al. (2003). Nonlinear set-theoretic localization of cellular phones. In *AeroSense 2003* (pp. 51–58). International Society for Optics and Photonics.
- Jaulin, L., Kieffer, M., Didrit, O., & Walter, É. (2001). *Applied interval analysis, with examples in parameter and state estimation, robust control and robotics*. Springer.
- Jaulin, L., & Walter, E. (1993). Set inversion via interval analysis for nonlinear bounded-error estimation. *Automatica*, 29(4), 1053–1064.
- Jirawimut, R., et al. (2003). A method for dead reckoning parameter correction in pedestrian navigation system. *IEEE Transactions on Instrumentation and Measurement*, 52(1), 209–215.
- Kieffer, M., Jaulin, L., Walter, É., & Meisel, D. (2000). Robust autonomous localization using interval analysis. *Reliable Computing*, 6(3), 337–362.
- Lazarus, S., et al. (2007). Vehicle localization using sensors data fusion via integration of covariance intersection and interval analysis. *IEEE Sensors Journal*, 7(9), 1302–1314.
- Martin, E., Vinyals, O., Friedland, G., & Bajcsy, R. (2010). Precise indoor localization using smart phones. In *Proceedings of the international conference on multimedia* (pp. 787–790).
- Moore, R. E., Kearfott, R. B., & Cloud, M. J. (2009). *Introduction to interval analysis*. SIAM.
- Qi, Y., & Kobayashi, H. (2003). On relation among time delay and signal strength based geolocation methods. In *Proceedings of IEEE global telecommunications conference* (pp. 4079–4083).
- Rappaport, T. (1996). *Wireless communications, principles and practice*. IEEE Press, Prentice Hall.
- Ristic, B., Arulampalam, S., & Gordon, N. (2004). *Beyond the Kalman filter: particle filters for tracking applications*. Artech House.
- Rump, S. M. (1999). INTLAB-INTERVAL LABORATORY. In Tibor Csendes (Ed.), *Developments in reliable computing* (pp. 77–105). Dordrecht, Netherlands: Kluwer.
- Sayed, A. H., & Tarighat, A. (2005). Network-based wireless location, challenges faced in developing techniques for accurate wireless location information. *IEEE Signal Processing Magazine*, 22(4), 24–40.
- Schweppe, F. C. (1973). *Uncertain dynamic systems*. Prentice-Hall.
- Weiss, A. (2003). On the accuracy of a cellular location system based on RSS measurements. *IEEE Transactions on Vehicular Technology*, 52(6), 1508–1518.



Yan Cui received his B.S. degree in Mechanical Engineering at Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2009, and M.S. degree in Mechanical Engineering at Purdue University, West Lafayette, Indiana, in 2011, where he is now pursuing Ph.D. degree in the area of systems, measurement and control. He is working in the Autonomous Systems Laboratory in the school and currently developing novel algorithms to improve cell phone geolocation. His research interests include optimal estimation, system identification, and pedestrian navigation systems.



Ruxiao An received the B.S. degree in Mechanical Engineering from Purdue University, West Lafayette, Indiana, in 2013, and the B.S. degree in Mechanical Engineering from Shanghai Jiao Tong University, Shanghai, China, in 2013. He has been working in the Autonomous Systems Lab at Purdue University as an undergraduate research assistant since 2011. In the summer of 2012, he was awarded with Purdue Summer Undergraduate Research Fellowship. He is currently pursuing M.S. degree in Mechanical Engineering at Stanford University, Stanford, California.



Kartik B. Ariyur is Assistant Professor of Mechanical Engineering at Purdue University. He runs the Autonomous Systems Laboratory in the school. His group works on several aspects of autonomous operation from sensor design and characterization to multi-vehicle mission planning and health monitoring. Prior to joining Purdue, he worked at the Honeywell Labs in Minneapolis where he worked on a range of problems in guidance, navigation, control, health monitoring, system verification and surveillance. His health monitoring algorithms currently run on Honeywell APUs in service in commercial aircraft (around 70% of all commercial aircraft); his front end filtering algorithms developed at Qualcomm reside in all CDMA chips in cell-phones. He has authored 16 peer reviewed journal papers, 48 conference papers, and 17 US patents, and another 15 US patents pending. He co-authored the text, 'Real-time Optimization by Extremum Seeking Control', published by John Wiley and Sons, 2003. He is currently Associate editor of the International Journal of Adaptive Control and Signal Processing, and member of the IEEE-CSS Conference Editorial Board. He has served on the program committees of the ACC, the IEEE CDC, HSCC (Hybrid Systems Computation and Control), and IEEE MSC (Multi-Systems Conference).