Cost-effective, asynchronous inter-sensor distance estimation using trigonometry

J.-W. Yoon, H. Park, N.S. Kim and T. Park✉

Precise estimation of inter-sensor distances is essential for reliable localisation in Internet of Things sensor networks. A cost-effective, scalable, asynchronous solution to estimate inter-sensor distances based solely on measurements of distances to a moving object is proposed. More specifically, the proposed solution estimates uncharted distances using trigonometry and processes these estimated distances with a distributed weighted multi-dimensional scaling algorithm for more precise localisation of sensors. It is demonstrated that the proposed solution achieves the localisation error of 4.8–33.9 cm when the measurement errors of sensor devices are in the range of 5–40%.

Introduction: The advent of Internet of Things (IoT) enables physical things to sense and control environments remotely and ubiquitously. As the cyber world gets tightly coupled with the physical world, IoT services can potentially offer huge economic benefits. To reap such benefits, many IoT applications require exact localisation based on distances amongst things, i.e. IoT/sensor devices. Most prior localisation techniques have assumed that these distances are always known or given in advance [1]. However, as IoT devices typically use cheap off-the-shelf heterogeneous sensors and/or line-of-sights can be easily blocked by obstacles, distances amongst IoT devices (i.e. inter-sensor distances) are likely to be unavailable or partially available only for certain area [2]. In such cases, applications relying on localisation through precise inter-sensor distances are likely to make poor or wrong decisions, which diminishes the usefulness of such applications.

To estimate inter-sensor distances, we may consider some known techniques such as time difference of arrival (TDOA) and received signal strength indication (RSSI) that rely on transmission/reception of RF signals. Nevertheless, it is impractical to use TDOA since tight clock synchronisation amongst IoT devices is very challenging for resource-poor devices [3]. Moreover, RSSI cannot satisfy the required level of accuracy in estimating inter-sensor distances.

Motivated by this, we propose *trigonometry-based inter-sensor distance estimation* (TIDE), which *estimates* distances between IoT devices solely based on measuring distances from IoT devices to a *moving object* (e.g. a resident in a smart home) with ultrasound, infrared, or laser sensors; IoT devices are typically too small for cheap ultrasound, infrared, or laser sensors (without RF capability) to measure distances to other small IoT devices. For precise estimation of inter-sensor distances using these cheap sensors, TIDE (i) identifies on-the-fly *triangles* amongst devices, each enclosing the moving object, by using trigonometry; (ii) estimates an inter-sensor distance between two devices belonging to each triangle; and (iii) integrates/averages these inter-sensor distances to improve the estimation accuracy. Subsequently, the estimated inter-sensor distances can be used by a localisation algorithm to construct a network-wide map of device locations.

To choose a suitable localisation algorithm, we must consider two challenges: (i) only a subset of devices within the network can measure distances to a moving object at a certain point of time, and (ii) the estimated inter-sensor distances may still render some errors. To address these two challenges, we synergistically incorporate TIDE with the distributed weighted multi-dimensional scaling (dwMDS) algorithm [4]. A salient aspect of TIDE with dwMDS (TIDE/dwMDS) is that we can achieve high precision for both inter-sensor distance estimation and network-wide localisation although the network does not offer clock synchronisation amongst devices. Our evaluation shows that TIDE/dwMDS can achieve the inter-sensor distance estimation error of 6.17 and 30.45 cm for distance measurement error of 5 and 25%, respectively. In subsequent sections, we describe the detailed procedure of TIDE/dwMDS and evaluate its efficacy.

Trigonometry-based inter-sensor distance estimation: Given a network of N IoT devices, $S = \{s_i\}_{i=1}^{N}$, we first determine a set of triangles at time $t > 1$, $\Gamma(t)$, that *encloses a moving object* using a trigonometrical procedures as follows. We first find a set of devices that has detected the moving object at time $t-1$ and t. For each device s_i that has detected the moving object, we compute the difference of measured distances $\Delta d_i(t) = d_i(t) - d_i(t-1)$, where $d_i(t)$ and $d_i(t-1)$ are measured distances between s_i and a moving object at time t and $t-1$, respectively. Furthermore, we group s_i with other two neighbouring devices to form a unique triangle. If all of $\Delta d_i(t)$ constituting the triangle are either positive (increasing) or negative (decreasing), the mobile object is concluded to be outside the triangle. In contrast, the mobile object is enclosed within the triangle if one of $\Delta d_i(t)$ has a sign opposite to the rest, and hence we add that triangle to $\Gamma(t)$ while the device with an opposite sign from the other two is later used as a pivot device for inter-sensor distance estimations.

Once $\Gamma(t)$ is identified, we estimate the inter-sensor distances between devices s_i and s_j, where $i \neq j$ and $1 \leq i, j \leq N$, as follows. First, we identify a set of triangles $\bar{\Gamma}_{ij}(t) \subset \Gamma(t)$ satisfying two conditions: (i) each triangle has s_i and s_j as vertices, and (ii) s_i or s_j must be the pivot device. This set may or may not be empty. If $\bar{\Gamma}_{ij}(t)$ is empty, inter-sensor distance estimation between s_i and s_j cannot be made at time t. On the other hand, when the set is not empty, the inter-sensor distance between s_i and s_j, $\delta_{ij}^k(t)$, is computed using the kth triangle of $\bar{\Gamma}_{ij}(t)$ as follows

$$\begin{aligned}\delta_{ij}^k(t) &\simeq \frac{\mathcal{LB}_{ij}^k(t) + \mathcal{UB}_{ij}^k(t)}{2} \\ &= \frac{\sqrt{d_i(t)^2 + d_j(t)^2} + d_i(t) + d_j(t)}{2}\end{aligned} \tag{1}$$

where $\mathcal{LB}_{ij}^k(t)$ and $\mathcal{UB}_{ij}^k(t)$ distances are illustrated in Fig. 1*a*, which are limited by the angles ranging from 90 to 180° due to the trigonometry and the aforementioned enclosing triangle detection mechanisms. Note that multiple triangles within $\bar{\Gamma}_{ij}(t)$ which include s_i and s_j at time t can generate multiple distance estimations of $\delta_{ij}^k(t)$.

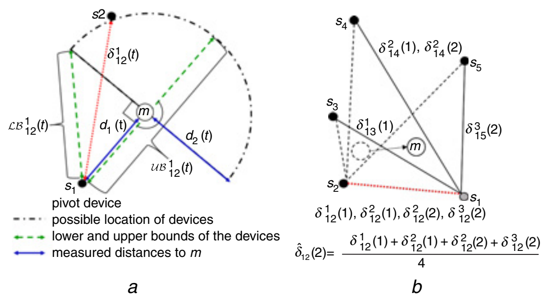

Fig. 1 *Examples of TIDE*

a Relative distance measurements amongst neighbour devices
b Computation of $\hat{\delta}_{ij}(t)$ when object moves

Algorithm 1 TIDE and dwMDS pseudocode

```
Initialise:
    t = 0; random coordinates X(0) = [x_1(0), …, x_N(0)];
    C(0); compute a_i according to [4]
do
    t ← t + 1
    for k = 1 to |Γ̄_ij(t)| do
        for all i and j combination do
            Compute δ^k_ij(t) using (1)
    if t ≥ K then
        for all i and j combination do
            Compute δ̂_ij(t) using (2)
        C(t) = C(t − 1)
        for i = 1 to N do
            Compute b_i(t − 1) according to [4]
            x_i(t) = a_i X(t − 1) b_i(t − 1)
            Compute c_i(t) using (3)
            C(t) ← C(t) − c_i(t − 1) + c_i(t)
            Broadcast x_i(t) to neighbours of s_i and send C(t) to the next device
while C(t − 1) − C(t) < ε
```

To further improve the accuracy, we integrate $\delta_{ij}^k(t)$ for the past K rounds of estimations and compute an average value as

$$\hat{\delta}_{ij}(t) = \frac{\sum_l \sum_{k=1}^{|\bar{\Gamma}_{ij}(l)|} \delta_{ij}^k(l)}{\sum_l |\bar{\Gamma}_{ij}(l)|} \tag{2}$$

where $t \geq K$, $t - K + 1 \leq l \leq t$, and $|\bar{\Gamma}_{ij}(l)|$ is the number of estimations (triangles) made between s_i and s_j at iteration l. Running example can be found in Fig. 1*b*. Once $\hat{\delta}_{ij}(t)$ is acquired by going through the above processes, this information is applied to dwMDS to identify the precise coordinates of devices.

According to our experiment, TIDE can offer reasonably precise distance estimation results. This indicates that TIDE can be applied to a variety of IoT network applications without the need for explicitly measuring inter-sensor distances.

Self-localisation through dwMDS: Given $\hat{\delta}_{ij}(t)$, we apply dwMDS to perform localisation. Considering that IoT environments foster distributed and heterogeneous network of cheap, off-the-shelf IoT devices, dwMDS may satisfy such needs as device coordinates are iteratively identified using $\hat{\delta}_{ij}(t)$. Furthermore, we note that the effect of errors in distance estimation by TIDE can be gracefully resolved using dwMDS since iterative minimisation of the global cost can adjust the coordinates of individual device at every iteration based on their distance relationship between itself and the neighbours.

Let $\boldsymbol{x}_i(t)$ denote a $p \times 1$ coordinate vector representing an estimated coordinate of device s_i at iteration t. The dwMDS searches s_i's true coordinate by iteratively minimising the global cost, $C(t)$ as depicted in Algorithm 1, in which it: (i) randomly chooses its initial coordinate matrix, $\boldsymbol{X}(0) = [\boldsymbol{x}_1(0), \ldots, \boldsymbol{x}_N(0)]$; (ii) at each iteration t, refines the estimated coordinates by processing $\{\hat{\delta}_{ij}(t), \boldsymbol{x}_i(t-1)\}_{j \neq i}$ with update formulae for a_i and $\boldsymbol{b}_i(t)$ defined in [4]; (iii) broadcasts the updated coordinate $\boldsymbol{x}_i(t)$ to neighbours of s_i as well as send $C(t)$ to the next device; and (iv) terminates its process when the estimated coordinate converges to a certain location such that $C(t)$ is minimised below a threshold, i.e. $C(t-1) - C(t) < \varepsilon$, and otherwise, repeats the process at the next iteration.

The local cost between s_i and s_j in the tth iteration is defined as $c_{ij}(t) = w_{ij}(\hat{\delta}_{ij}(t) - \|\boldsymbol{x}_i(t) - \boldsymbol{x}_j(t)\|)^2$ where $\|\boldsymbol{x}\|$ is the Euclidean norm of the vector $\boldsymbol{x}$, and the weight w_{ij} reflects the accuracy of estimated distances between s_i and s_j. The cost of s_i, $c_i(t)$, at iteration t is then computed by summing up the local costs as

$$c_i(t) = \sum_{j=1}^{N} c_{ij}(t) = \sum_{j=1}^{N} w_{ij}\left[\hat{\delta}_{ij}(t) - \|\boldsymbol{x}_i(t) - \boldsymbol{x}_j(t)\|\right]^2 \qquad (3)$$

Performance evaluation: The efficacy of TIDE/dwMDS is evaluated under two different network topology settings: (i) devices form a square grid with an edge length of 1 m, and (ii) devices are randomly deployed. Throughout the evaluation, a mobile object is randomly moving at the speed of 4 km/h. We add uniformly generated random errors to distance measurements, e.g. a 5% error in measuring 1 m distance produces random values in the range of 0.95 and 1.05 m. We also use two performance criteria: the accuracy of inter-sensor distance estimations and the error of localisation.

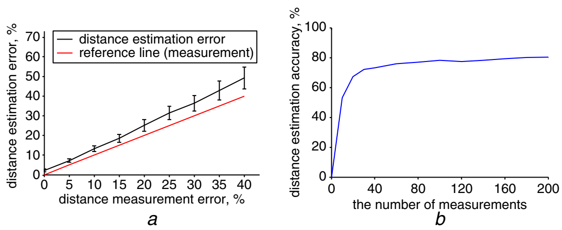

Fig. 2 *Evaluation of TIDE*

a Inter-sensor distance estimation error versus error in real measurements
b Inter-sensor distance estimation accuracy versus number of measurements from IoT devices

To evaluate the accuracy of inter-sensor distance estimations, we create a simulation environment consisting of four devices forming a square grid. As shown in Fig. 2*a*, the distance estimation error is linearly proportional to the measurement error added to $d_i(t)$. When the measurement error equals 20%, TIDE/dwMDS achieves the estimation error of 24.1% with standard deviation of 4.2. Furthermore, the measurement error of 40% yields the estimation error of 48.2% with standard deviation of 7.9. In comparison with actual measurement of distances between devices, TIDE/dwMDS yields marginal distance estimation errors. Considering that an additional distance measurement error of 20% can be easily observed in real environments, we may determine the number of needed measurements to achieve reasonable estimation accuracy using $\hat{\delta}_{ij}(t)$. According to Fig. 2*b*, the estimation accuracy of 75.9% is achieved when 25 or more measurements are made. Increasing the number of measurements beyond 50 has a marginal effect on estimation accuracy, and the estimation error converges to the desired reference after 180 or more measurements.

For the evaluation of localisation performance, we generate two different network topologies, grid style and random deployment, by varying the number of devices, N as plotted in Fig. 3. The results in Fig. 3 shows that increasing the number of devices lowers the localisation error due to the fact that the number of triangles that include s_i and s_j increases as well. In case of a grid-like network topology, $N > 16$ shows marginal gain in estimation accuracy, while random deployment achieves a smaller gain when $N > 25$. The insight behind such marginal gains in estimation accuracy with more devices is that however big the network is, the number of triangles including s_i and s_j is limited by the range of the distance measurement used by the devices. The evaluation of estimation accuracy plotted in Fig. 3 indicate that the achievable localisation error can be <10 cm when the measurement error rate is at 20% and $N \geq 9$. After analysing the estimation accuracy of TIDE/dwMDS, we conclude that our proposed algorithm is suitable to estimate the inter-sensor distances for IoT environments where direct device-to-device distance measurements are not possible or available.

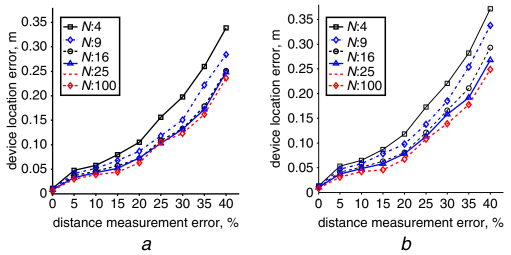

Fig. 3 *Simulation results of location estimation using dwMDS*

a Fixed, grid-style topology
b Random topology

Conclusion and future work: We propose TIDE using the distance measurements between the IoT devices and the moving object only. Our solution is designed to support distributed systems by fostering asynchronous distance estimation algorithms. The inter-sensor distance estimation results are fed to a dwMDS localisation algorithm to identify relative sensor coordinates. Performance evaluation results indicate that TIDE is able to achieve reasonably accurate inter-sensor distance estimation and localisation results. We note that TIDE may suffer from geometrical flex ambiguities, which will be address in the future works. Furthermore, we will deploy our solution into real environments to conduct comprehensive experiments, and make improvements.

Acknowledgment: This work was supported by the research fund of Hanyang University (HY-2015-G).

Submitted: *5 February 2016* E-first: *11 May 2016*
doi: 10.1049/el.2016.0442
One or more of the Figures in this Letter are available in colour online.

J.-W. Yoon and H. Park (*Department of Information and Communication Engineering, DGIST, Daegu, Republic of Korea*)

N.S. Kim (*Department of Electrical and Computer Engineering, University of Illinois, Urbana-Champaign, IL, USA*)

T. Park (*Department of Robotics Engineering, Hanyang University, Gyeonggi-do, Republic of Korea*)

✉ E-mail: taejoon@hanyang.ac.kr

References

1 Tian, H., Chengdu, H., Brian, M.B., John, A.S., and Tarek, A.: 'Range-free localization schemes for large scale sensor networks'. ACM MobiCom'03, San Diego, CA, USA, September 2003, pp. 81–95
2 Guoping, C., and Li, W.: 'Precise indoor localization technology based on TR-FMM', *Electron. Lett.*, 2014, **50**, pp. 1248–1250
3 Park, H., Park, J., Kim, H., *et al.*: 'ReLiSCE: utilizing resource-limited sensors for office activity context extraction', *IEEE Trans. Syst. Man. Cybern., Syst.*, 2015, **45**, pp. 1151–1164
4 Jose, A.C., Neal, P., and Alfred, O.H.III: 'Distributed weighted-multidimensional scaling for node localization in sensor networks', *ACM Trans. Sensor Netw.*, 2006, **2**, pp. 39–64