Accurate Indoor Positioning Methods for Smart Devices Using Improved Pedestrian Dead-Reckoning and Collaborative Positioning Techniques

Liew Lin Shen

Faculty of Engineering, Computing & Science
Swinburne University of Technology
Sarawak Campus, Malaysia

This dissertation is submitted for the degree of

Doctor of Philosophy

March 2017

Dedicated to ...

my beloved Parents & Family

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other University. This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where specifically indicated in the text.

Liew Lin Shen
March 2017

Acknowledgements

Doing this PhD made me aware of my ignorance and, hopefully, I will always be reminded of the importance of lifelong learning and humility. This PhD journey of mine would have been unbearable without various forms of helps from many people, and therefore I would like to offer my sincerest thanks to all of them.

I am irredeemably indebted to my family especially my parents, for they have supported me both financially and morally for so long so that I could fully focus on my study.

I feel privileged that my PhD study was conducted under the supervision of Dr. Wallace Wong Shung Hui. His constant encouragement, illuminating guidance and editorial advice were indeed essential for the completion of this dissertation.

I also want to thank my friends and colleagues. We had memorable times, lots of fun, laughs, sharing of life stories, etc which undoubtedly added colors to my PhD journey.

Last but the most important, I would like to express my utmost gratitude to God Almighty for seeing me through all these years, for all the blessing He has showered upon me, and for the good times as well as the hard times.

Abstract

In view of arising demand for indoor Location-Based Services (LBS), many indoor positioning systems (IPS) have been developed by utilizing various technologies and yet none of them are gaining mainstream adoption. Perhaps the most promising IPS are those which are based on smart-devices, because the smart-devices are increasingly widespread and originally equipped with several useful sensors. This thesis aims at proposing accurate indoor positioning methods that leverage on ready infrastructure like smart-devices and Wi-Fi access points. Pedestrian Dead-Reckoning (PDR) is employed for its simplicity. In order to retain the relatively high positioning accuracy of PDR for long-term positioning, the Received Signal Strength Indicator values obtained from the site's Wi-Fi access points are used in two unique ways to mitigate the accumulative error of PDR. Besides, the estimated position of individual pedestrian can also be refined via an iterative process based on information derived from a Directed Graph. The Directed Graph is used to represent the relations among the pedestrians and the site's Wi-Fi access points. The proposed indoor positioning methods had been tested at real sites within Swinburne University of Technology Sarawak Campus, and were benchmarked against some existing indoor positioning methods. The test results imply that the proposed methods outperform some existing methods in terms of positioning accuracy.

Publications arising from this Work

Journal papers

- Liew, L. S. and Wong, W. S. H. (2016). Improved pedestrian dead-reckoning - based indoor positioning by rssi-based heading correction. *IEEE Sensors Journal*, 16(21):7762–7773

- Liew, L. S. and Wong, W. S. H.. Directed graph based collaborative indoor positioning method. *Elsevier Measurement* (In Review)

Conference paper

- Liew, L. S. and Wong, W. S. H. (2014). Indoor positioning method based on inertial data, rssi and compass from handheld smart-device. *2014 International Conference on Computer, Communications, and Control Technology (I4CT)*, pp. 48-52. doi: 10.1109/I4CT.2014.6914143

Book chapter

- Wong, W. S. H., Liew, L. S., Lai, C. H. and Liu, L. Accurate Indoor Positioning Technique Using RSSI Assisted Inertial Measurement. In: Jung, H. K., Kim, J. T., editors. Future Information Communication Technology and Applications: ICFICE 2013. Springer Netherlands; 2013. p. 121-129.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

AOA	Angle of Arrival
AP	Access Point
API	Application Program Interface
BN	Beacon Node
COO	Cell of Origin
DR	Dead-Reckoning
FD	Frequency Domain
FP	Fingerprinting
GPS	Global Positioning System
IPS	Indoor Positioning System
KF	Kalman Filter
LBS	Location-Based Services
LOS	Line-Of-Sight
MCMC	Markov Chain Monte Carlo
MEMS	Micro-Electro-Mechanical Systems
MN	Mobile Node
NLOS	Non-Line-Of-Sight

PDOA	Phase Difference of Arrival
PDR	Pedestrian Dead-Reckoning
PF	Particle Filter
RFID	Radio Frequency Identification
RP	Reference Point
RSSI	Received Signal Strength Indicator
RTT	Round Trip Time
SD	Spatial Domain
SIR	Sampling Importance Resampling
TD	Time Domain
TDOA	Time Difference of Arrival
TOA	Time of Arrival
TOF	Time of Flight
UWB	Ultra-Wideband

Chapter 1

Introduction

1.1 Motivation

The integration of Global Positioning System (GPS) technology into mobile devices like laptops and smart-devices has resulted in a tremendous growth of Location-Based Services (LBS) recently. LBS [66] can be defined as services that offer information, entertainment and security to the end user through a mobile device according to the device's geographical location – to name a few, providing navigation guidance for the user to reach desired destinations, streaming advertisements and coupons of nearby stores to potential customers, and tracking people and assets for security purposes.

The core of reliable LBS is to track the target's real-time location accurately. GPS has been commonly acknowledged as the most reliable location-sensing system for civilian use, which is able to achieve an accuracy up to 5 meters [27]. Unfortunately, GPS is only usable when there's Line-Of-Sight (LOS) connection to at least four satellites [26, 38], thus rendering it unreliable in indoor environments [24, 47] where obstruction of satellites signals due to obstacles like floors and walls can be very severe. Hence, the LBS to date are mostly developed for outdoors. However, as people nowadays spend most of their time in indoors [84] like multilevel offices, mega-malls, universities and transportation facilities, indoor LBS are getting rather essential.

In view of the great potentials of indoor LBS, indoor positioning has become an increasingly popular research topic. In fact, many solutions to indoor positioning, aka Indoor Positioning Systems (IPS), have already been developed by utilizing various technologies [16, 22, 47, 82, 84] encompassing cellular networks, Micro-Electro-Mechanical Systems (MEMS) sensors [24], Wi-Fi [80], Ultrasound [78], Ultra-Wideband (UWB) [4], Radio Frequency Identification (RFID) [9] and Bluetooth [85]. Though so, none of the existing IPS are considered de facto standard for indoor positioning due to their uniqueness in various

aspects such as accuracy and cost. While higher accuracy is naturally desirable for an IPS, its corresponding cost remains a major concern that affects its popularity. The cost of an IPS may depend on a variety of factors like resources and inconveniences, that are necessary for its deployment. Oftentimes, higher accuracy comes with higher cost as the hardware involved gets increased either in quantity or quality. For example, *Active Bat* [6] may be more accurate than *Active Badge* [78] as their accuracies are centimetre-level and room-level, respectively; but the former incurs higher cost because it requires comparatively greater amount of sensors to be installed at the site [22].

Smart-devices (e.g. smart-phones and tablets) and Wi-Fi Access Points (AP) have been widespread recently, thereby triggering a growing interest among researchers in utilizing these ready infrastructure in designing their IPS as an effort to minimize deployment cost while achieving sufficient positioning accuracy. Besides having wireless capabilities (e.g. Wi-Fi and Bluetooth), smart-devices are also equipped with a variety of sensors such as 3-axial accelerometers, gyroscopes and magnetometers, which can be harnessed to provide information about the user's body movements. Therefore, Pedestrian Dead-Reckoning (PDR) and Received Signal Strength Indicator (RSSI) based approaches namely Fingerprinting and Lateration are among the most popular positioning techniques employed in these smart-device based IPS. Certainly these techniques are not without limitations, yet they could individually be refined or combined with one another via appropriate use of filters, settings, etc to achieve even better performance in terms of accuracy and robustness.

1.2 Research Objective

The key objective of this research is to propose a new solution to estimating the location of smart-device-carrying pedestrian in a Wi-Fi enabled indoor environment, whilst meeting the following criteria:

- Minimum cost: In order to ease the adoption of an IPS by the mass of pedestrians, its associated cost ought to be kept minimum. Besides the resources (i.e. time, money and labor) that are necessary for IPS installation and maintenance, the cost includes the inconveniences that may be imposed on the user – for instance, requiring the users to carry additional and/or dedicated hardware that might restrict their usual body movements.

- Decent Accuracy: The metric used for evaluating the accuracy of an IPS is the discrepancy (error) between its estimation result and pedestrian's actual position. The lower the discrepancy, the higher the accuracy. Higher accuracy is of course more desirable, however compromises the cost. As the emphasis is placed on estimating the indoor position of pedestrian for general LBS purposes, IPS that is able to tell the pedestrian's indoor location with meter-level accuracy should be sufficient since its outdoor counterpart (i.e. GPS) has a similar level of accuracy as well. Centimeter-level accuracy seems superfluous in indicating the pedestrian's position on a map that depicts the building's indoor layout and segmentation (e.g. floors, rooms and walkways) at reasonable resolution.

- Decent Robustness: Robustness of an IPS can be defined as its resilience and ability to compute the pedestrian's location despite the information obtained for computation is incomplete. Information may be incomplete at times because of signal loss for some reasons. For example, sole Wi-Fi -based Tri-Lateration shall fail entirely when less than three Wi-Fi sources are available for provision of necessary signals. Therefore, seamless cooperation between different positioning techniques or technologies is more favored for the sake of higher robustness.

1.3 Problem Statement

On the basis of research objective, the proposed solution is confined to leverage only the ready infrastructure (i.e. smart-device and Wi-Fi access points) without introducing additional hardware.

Indeed, there is a considerable amount of existing works that have utilized the same infrastructure. Most of them were inspired thanks to a variety of sensors (e.g. accelerometers, gyroscopes and magnetometers) originally embedded on the smart-device that enable the implementation of PDR. PDR is attractive for being a stand-alone solution that estimates the pedestrian's position by detecting their step and heading based on data solely extracted from the specified sensors. Unfortunately, these sensors especially the low-cost ones like smart-device's are bound to suffer from inherent biases and drifts, thereby jeopardizing PDR's long-term performance.

Alternatively, since the smart-device has the wireless capabilities to communicate with surrounding Wi-Fi or Bluetooth counterparts, conventional positioning techniques such as Lateration and Fingerprinting could be viable by making use of the wireless signals. Typically, RSSI values of detected signals are measured and assumed to correspond the transceivers' positions or separation distances based on theoretical or empirical models. However, the assumption itself is barely practical because these radio signals are notorious for being erratic at times owing to radio multipath, reflection, etc, especially in indoor environments where obstacles are plenty. Nevertheless, the aforementioned positioning techniques have their merits, and therefore the challenge lies in resolving their associated drawbacks in order to achieve reliable indoor positioning.

1.4 Contributions

The contributions of this work are:

- proposes a scheme that refines PDR-estimated position by making use of relatively stronger RSSI as well as compass bearing.
- proposes a scheme that utilizes RSSI to correct PDR-estimated heading based on linear regression.
- findings showing that the proposed schemes outperform some existing PDR-based indoor positioning methods in terms of accuracy and robustness.
- uses Directed Graph to represent the relations among pedestrians.
- proposes a collaborative positioning method that derives information from the Directed Graph to improve the positions estimated by PDR or Tri-Lateration via a Particle Filter -based iterative process.

1.5 Thesis Structure

The rest of this thesis is organised into six chapters. **Chapter 2** summarizes the fundamentals of indoor positioning as well as some related works. **Chapter 3** describes the setup and preliminaries that were necessary for the evaluations of proposed methods. **Chapter 4** discusses the two proposed schemes that improve PDR-estimates. **Chapter 5** discusses the proposed collaborative method that refines position estimates based on Directed Graph and Particle Filter. Finally, **Chapter 6** summarizes this thesis and provides some directions for the future work.

Chapter 2

Background

This chapter gives an overview over the fundamentals of indoor positioning by discussing the major types of signal measurements which are used to convey the geometric relation between wireless sensors, and conventional positioning techniques including PDR and Lateration. In addition, Bayesian filters namely Kalman filter and Particle filter are also briefly described, as they have been widely employed to fuse multiple information for optimal estimates. Lastly, a summary of some related works is also presented. Portions of the material presented in this chapter had been published [45].

2.1 Geometric Measurements with Wireless Technologies

Various wireless technologies including RFID and Ultrasound have been favored among the existing IPS, because their corresponding wireless transmissions could be useful in deriving measurements that tell the geometric relation between the sensors. Nevertheless, the derived measurements are fallible because the wireless signals (i.e. electromagnetic waves and sound waves) are susceptible to wave phenomena (e.g. multipath, interference, reflection and refraction) especially in indoor environments where obstacles made of various materials are present. Some major geometric measurements based on wireless transmission are described here. These geometric measurements provide core estimates (e.g. distance or angle) that are required by certain conventional positioning techniques in computing the position. Note that *beacon* node denotes the sensor which location is known, and *mobile* node represents the sensor held by the moving object of interest (i.e. pedestrian).

Time of Arrival (TOA)

TOA, also known as Time of Flight (TOF), is the time taken for a radio signal transmitted from node A to arrive at node B. While measuring the TOA, the involved nodes must be tightly synchronized, and the time-stamp of transmission signal is crucial. The distance between the two nodes can be calculated by simply multiplying the measured TOA with speed of the wave.

Time Difference of Arrival (TDOA)

TDOA is the time difference between the two TOAs, which can be measured in two different ways: firstly, a reference signal broadcast by a *mobile* node to reach a pair of *beacon* nodes; secondly, signals emitted by a pair of *beacon* nodes to reach a *mobile* node [16]. In either ways, the *beacon* nodes have to be in sync. An TDOA results in a hyperbola that represents the likely positions of the *mobile* node (say m) in relative to the two *beacon* nodes (say i and j), based on the equation [47]:

$$d_{i,j} = \sqrt{(x_i - x_m)^2 + (y_i - y_m)^2 + (z_i - z_m)^2} - \sqrt{(x_j - x_m)^2 + (y_j - y_m)^2 + (z_j - z_m)^2} \tag{2.1}$$

where $d_{i,j}$ denotes the constant range difference which can be calculated by multiplying the measured TDOA with speed of the wave; x, y and z denote the coordinates in three dimensional space.

Round Trip Time (RTT)

RTT is the time required for a signal sent by node A to arrive at node B followed by an acknowledgement signal replied by node B after a response delay to reach node A. So, it's basically a summation of two TOAs and a response delay. As RTT is measured by node A by keeping track of the times it sends the signal as well as receives the acknowledgement, this eliminates the necessity of having synchronization like that of TOA measurement. However, the difficulty lies in measuring the exact response delay which ideally has to be excluded from RTT in order to derive the distance between the nodes. Nonetheless, the delay could be ignored when it's substantially smaller than the total transmission time [47]. The distance between the two nodes is simply the multiplication between halved RTT and speed of the wave.

Phase Difference of Arrival (PDOA)

There are three main variants of PDOA, and they can be distinguished as Time Domain (TD), Frequency Domain (FD) and Spatial Domain (SD) [55]. Note that the phase of the signal received by the reader is expressed as:

$$\phi = \phi_p + \phi_o + \phi_b \tag{2.2}$$

where ϕ_p denotes the phase accumulated due to electromagnetic propagation, ϕ_o denotes the phase offset caused by hardware like cables and antenna components, and ϕ_b denotes the backscatter phase from the *mobile* node.

TD-PDOA is the difference between the phases received by the *beacon* node at two different time instances (say t_1 and t_2), which can be used to derive the *mobile* node's radial velocity (say V_r), as expressed by the equation:

$$\text{Radial velocity, } V_r = \frac{c}{4\pi f}\frac{(\phi_{t_2} - \phi_{t_1})}{t_2 - t_1} \tag{2.3}$$

where c is the light's speed and f is the signal's frequency. Nonetheless, the assumption is that the *mobile* node's moving speed, ϕ_o and ϕ_b are constant during the corresponding interval.

FD-PDOA is the difference between the phases received by the *beacon* node at two different frequencies (say f_1 and f_2), which can be used to estimate the distance between *mobile* node and *beacon* node, as expressed by the equation:

$$\text{Distance, } d = -\frac{c}{4\pi}\frac{(\phi_{f_2} - \phi_{f_1})}{f_2 - f_1} \tag{2.4}$$

Yet, the assumption is that the *mobile* node, ϕ_o and ϕ_b remain stationary during the FD-PDOA measurement.

SD-TDOA is the difference between the phases received by two different antennas at the same frequency channel, which can be used in deriving Angle of Arrival (AOA) as expressed by the equation:

$$\text{AOA, } \theta \approx \arcsin\left[-\frac{c}{2\pi f}\frac{(\phi_b - \phi_a)}{D}\right] \tag{2.5}$$

where D is the distance between two antennas on the antenna array. A simple illustration for this is given in Figure 2.1. However, the assumption is that the transmitter is considerably far from the antenna array, and signals received at the antennas are parallel. Further discussions on AOA estimation algorithms using antenna array can be found in [17, 39].

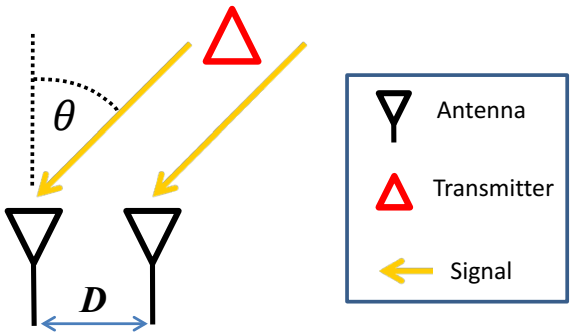


Fig. 2.1 Derivation of AOA from SD-TDOA.

Received Signal Strength Indicator (RSSI)

RSSI is a measure of power still present in the propagated radio wave when it is detected by a receiver. The RSSI value is typically represented in dBm, on a scale particular to the hardware chip-set. Greater RSSI value means shorter distance between transmitter and receiver. Several theoretical and empirical models [50, 58, 65, 67] could be used to express the correlation between RSSI and distance, and one of popular ones is shown below:

$$R = \beta - 10n\log_{10} d + \varepsilon \tag{2.6}$$

where R denotes the RSSI obtained at distance d, β denotes the RSSI measured at a reference distance which is usually 1 meter from the transmitter, n denotes the path loss exponent which typically ranges from 2 to 6 for indoor environments, and ε denotes a Gaussian random variable with zero mean and standard deviation σ_{rssi}. However, as the models assume that RSSI varies logarithmically with increasing distance in any directions from the source, their reliability are limited especially in indoor environments due to Non-Line-Of-Sight (NLOS) conditions as well as other factors like hardware specifications (e.g. chipset and antenna) and antenna's orientation.

2.2 Conventional Positioning Techniques

Proximity

This technique localizes the *mobile* node simply by binary connectivity between the *mobile* node and the *beacon* node. The *beacon* nodes are usually uniformly distributed over the area of interest. When the *mobile* node senses a *beacon* node via wireless communication, it is considered to be within the transmission range of corresponding *beacon* node. In case when two or more *beacon* nodes are detected simultaneously, only the one which contributes the strongest signal strengths shall be referred. However, the technique's limitation is that it tells not the absolute position (i.e. coordinates) but relative whereabouts of *mobile* node. For example, Cell Identification or Cell of Origin (COO) is a real-life application of this technique that tells which zone the cellular telephone lies within by identifying its associated cell tower. For indoor scenarios, the estimation resolution can be improved by covering the site of interest with higher density of *beacon* nodes which are preferably low-cost such as RFID tags [9].

Lateration

The concept of this technique is to deduce the *mobile* node's position by knowing how far it is apart from the reference node. The separation distance can be derived from measurements like RSSI, TOA, TDOA, RTT and PDOA. With single separation distance known, the node's position may be any point on the circle of the corresponding reference node as illustrated in Figure 2.2.

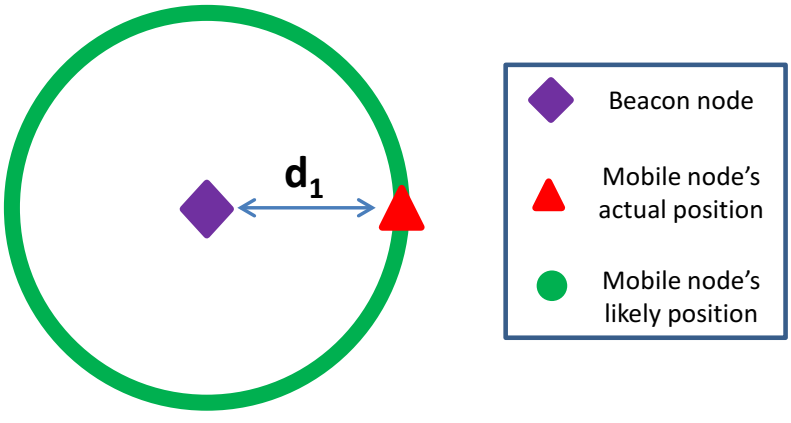


Fig. 2.2 Illustration: Lateration

For 2-dimensional positioning, as illustrated in Figure 2.3, three separation distances are typically used to find the intersection which signifies the *mobile* node's likely position, and

such procedure is particularly known as Tri-Lateration. Note that circles shall be replaced by hyperbolas when separation distances are derived from TDOAs, and the reference nodes must be non-collinear and non-collocated. When more than three separation distances are considered, 3-dimensional positioning is made possible and the procedure is called Multi-Lateration instead. In reality, the intersection is likely to happen at multiple points or may not happen at all, due to wave phenomena like multipath and absorption that result in poorly estimated separation distances. Therefore, the absolute position is usually finalized by finding one exact point which fulfills all the measured separation distances simultaneously in a Least-Squares sense [37, 62]. The estimation accuracy improves when more separation distances get taken into consideration. GPS technology is one prominent example that employs Multi-Lateration with TOA or TDOA based distance measurements [70].

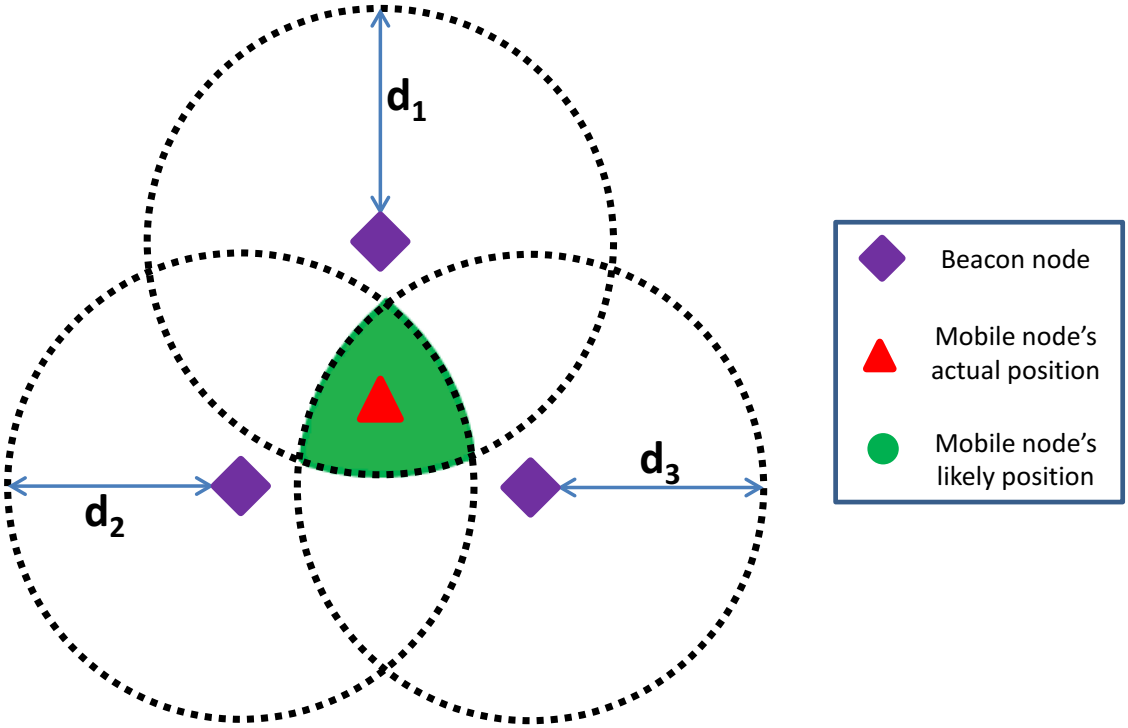


Fig. 2.3 Illustration: Tri-Lateration

Triangulation

An AOA only implies that the *mobile* node may lies on anywhere along the line heading in that specific direction from the reference, as depicted in Figure 2.4. However, the use of two AOAs at a time enables one to pinpoint the *mobile* node's 2-dimensional location. As illustrated in Figure 2.5, a triangle can be visualized by an intersection of two AOAs from different reference nodes whose positions are known; hence, the *mobile* node's can be deduced by finding the missing sides and angles of the triangle (based on trigonometry). Such procedure is known as Triangulation, otherwise called Multiangulation when more than two AOAs are considered during the computation. [68]. The latter enables 3-dimensional positioning. More AOAs considered means better estimation accuracy. However, the drawback of this technique is its complexity and deployment cost due to the use of directional antennas or antenna arrays for AOA measurements [4, 46]. Nevertheless, both Triangulation and Multiangulation have a long history in robot localization, and a summary of their diverse algorithms can be found in [59].

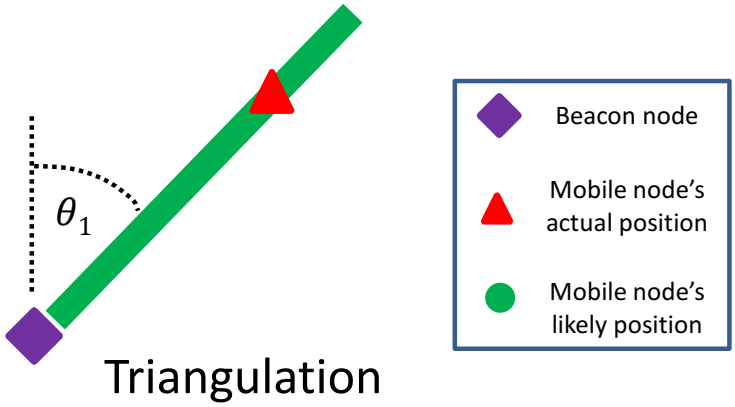


Fig. 2.4 Illustration: Single AOA

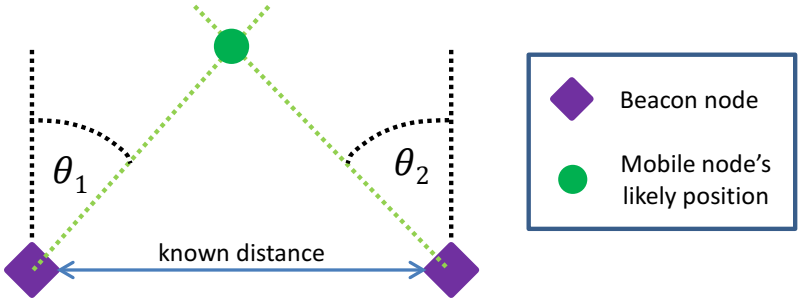


Fig. 2.5 Illustration: Triangulation

Fingerprinting

The underlying concept of Fingerprinting (FP) is that each particular location within the site of interest has a unique *fingerprint*. The *fingerprint* is actually a compilation of measurements which may be derived from radio signals (e.g.RSSI) from surrounding *beacon* nodes, magnetic field[10, 16, 60] and any other information that are location-dependent. Specifically, the procedure of Fingerprinting consists of two phases: offline phase and online phase. The offline phase serves to prepare the database prior to the online phase. In doing so, the site of interest is first divided into sub-areas and each sub-area is represented by a Reference Point (RP). At each RP, signals from deployed *beacon* nodes or related sensors are sampled multiple times in compiling a *fingerprint*. All *fingerprints* are then stored in a database along with position coordinates of their corresponding RPs. In the online phase, the real-time measurements by *mobile* node are used to perform matching with the *fingerprints* existing in the database, and hence coordinates of RP whose fingerprint yields the closest match shall be the estimated position of *mobile* node. Some typical matching algorithms are k-Nearest Neighbour (KNN), probabilistic methods, neural network, Support Vector Machine (SVM) and smallest M-vertex polygon (SMP) [47]. More details on a variety of FP methods can be found in [54].

Besides employing the right matching algorithm, the estimation accuracy improves by distributing more RPs over the site of interest. More RPs means each RP's denoted area shrinks. However, the problem is that sometimes the measurements obtained at some RPs might be very similar thereby causing their *fingerprints* hardly distinguishable. Therefore, the allocation of RPs must be carefully done for optimal precision. Given a fixed amount of RPs, FP's reliability can be improved by taking more *beacon* nodes into consideration [7, 30]. Though FP is well known for having high resilience towards NLOS conditions, the major downside is its laborious and time-consuming site-survey for database start-up as well as maintenance which is necessary to account for any changes made to the environment (e.g. addition/removal of *beacon* nodes or walls). Even so, FP has been popular among the Wi-Fi based IPS [25].

Dead-Reckoning

Unlike all the aforementioned positioning techniques, Dead Reckoning (DR) is a self-contained solution that requires no *beacon* nodes but only a single device like Inertial Measurement Unit (IMU) which contains all the related sensors to be carried by *mobile* node. The current position of *mobile* node is estimated by projecting forward its previous position with known distance and heading. For 2-D scenario, the *mobile* node's position P at time t can be simply computed by:

$$P_t = [X_t \;\; Y_t]^\intercal \tag{2.7a}$$

$$X_t = X_{t-1} + D.\cos(\theta_t) \tag{2.7b}$$

$$Y_t = Y_{t-1} + D.\sin(\theta_t) \tag{2.7c}$$

where X and Y denote coordinates in two dimensional space, θ is the heading and D is the distance traveled within the time interval. The very beginning position and heading of *mobile* node are often assumed known. Ideally, the consecutive positions can always be derived by knowing the distance traveled and heading at following time intervals. There are two ways to find the distance traveled: firstly, by double integration of successive acceleration measurements; secondly, by detecting the number of steps by observing certain patterns within the sensor data; the length of each step is either assumed to be constant [15, 49, 71] or dependent on various parameters such as step frequency and acceleration variance [13, 43, 63]. Specifically, DR that employs step detection in estimating the distance traveled is regarded as Pedestrian Dead-Reckoning (PDR). The heading is determined based on relative angular displacement [77] or absolute direction derived from either individual or fusion of readings from sensors like accelerometers, gyroscopes and magnetometers. However, as the sensors inherently suffer from non-zero and non-Gaussian noises, the resultant estimations are prone to accumulative errors thereby deteriorating DR's reliability over time.

2.3 Bayesian Filtering

The measurements taken by the sensors are never perfect, and so the estimates based on them are bound to have random noises. Fortunately, if prior knowledge about the system's dynamics and the involved sensors is available, then Bayesian Filtering [12] can be applied to merge them with the additional measurements/observations from other sensors to probabilistically estimate the system's states so that the error is minimized. In the context of indoor positioning, oftentimes the state of interest is the position of tracked object and the system's dynamics is how the object advances from its original positions to new ones based on every new sequence of available measurements/observations. Kalman Filter and Particle Filter are two most commonly used Bayesian filters. Generally, the state-space model of the system is expressed as follows:

$$x_t = g(x_{t-1}, u_{t-1}, w_{t-1}) \tag{2.8}$$

$$z_t = h(x_t, v_t) \tag{2.9}$$

where x is a vector representing the system's state at time t; the function $g(.)$ models the system's dynamics; z is the measurement otherwise known as measured state; the function $h(.)$ maps the state to a measurement; u is the control vector; and w and v denote the state's noise vector and measurement noise vector respectively.

Kalman Filtering

The Kalman Filter (KF) is a recursive means to optimally estimate the state of a linear dynamics system whereby the noises are Gaussian, so that the variance of estimation errors is minimum. There are some variations of Kalman Filter that are specifically designed for non-linear dynamics system, but consequently the variance of estimation errors has become approximately minimum instead [69]. Among the variations, Unscented Kalman Filter (UKF) and Extended Kalman Filter (EKF) are two popular ones. Regardless, the procedure of Kalman Filtering can be easily understood by firstly re-writing the model shown by Equation (2.8) and Equation (2.9) into a linear model as follows:

$$x_t = Ax_{t-1} + Bu_{t-1} + w_{t-1} \tag{2.10}$$

$$z_t = Hx_t + v_t \tag{2.11}$$

A, B and H are matrices; the process noise w and the measurement noise v are assumed independent of each other, and zero-mean Gaussian with covariance matrix Q and R respectively.

Once the KF model is defined, the KF algorithm that comprises two stages namely *Time Update* and *Measurement Update* shall be iterated at every new time step or whenever new measurement z_t is available. At every iteration, the stage *Time Update* comes first to obtain *a priori* estimates of the state x and the error covariance P at time t according to the equations

$$\hat{x}_t^- = A\hat{x}_{t-1} + Bu_t \tag{2.12}$$

$$P_t^- = AP_{t-1}A^\intercal + Q \tag{2.13}$$

where $\hat{x}_{t-1}$ and P_{t-1} denote the *a posteriori* estimates of the state and error covariance respectively at time $t-1$. The resultant *a priori* estimates namely $\hat{x}_t^-$ and P_t^- are then inputted to the stage *Measurement Update* to generate the corresponding *a posteriori* estimates by using the equations

$$K_t = P_t^- H^\intercal (HP_t^- H^\intercal + R)^{-1} \tag{2.14}$$

$$\hat{x}_t = \hat{x}_t^- + K_t(z_t - H\hat{x}_t^-) \tag{2.15}$$

$$P_t = P_t^- - K_t H P_t^- \tag{2.16}$$

where K denotes the Kalman gain.

If one thinks that the measurement z is more credible than the prediction (*a priori* estimate $\hat{x}^-$), then the covariance matrix R should be decreased for greater Kalman gain K, and hence greater credibility is given to z in computing the *aposteriori* estimate $\hat{x}$. An example of the implementation of KF in indoor positioning is presented in **Chapter 3.4.4**.

Particle Filtering

In contrast to Kalman Filter, Particle Filter (PF) is the optimal state estimator for non-linear dynamics systems whereby the noises are non-Gaussian [24]. However, it requires relatively higher computational time, which may be undesirable in certain cases where real-time estimates are needed. There are several variants of Particle Filter [5, 12], such as Sampling Importance Resampling (SIR) and Regularized Particle Filter (RPF). An example of SIR PF application is presented in Section 4.2.1. Despite the variants, the procedure of Particle Filtering can be generally described in 5 sequential steps:

1. Initially, a finite set of particles is generated to represent possible estimates of state x.
2. All particles are propagated according to a predefined state model.
3. Each particle is assigned a weight based on its similarity to the inputted measurements. Higher similarity results in greater weight. All weights are normalized.
4. A new set of particles is generated by drawing particles from existing set of particles in proportion to their assigned weights.
5. Iteration commences from step no.2 at every new time step or whenever new measurement is available.

Basically, the concept behind PF is to use a finite set of independent random variables called *particles* to approximately represent the posterior probability for any complex model (which is probably non-Gaussian and multi-modal), and the posterior probability is evolved with new measurements according to Bayes' theorem. An example of the implementation of PF in indoor positioning is presented in **Chapter 3.4.2**

2.4 Related Work

This work focuses on investigating ways that complement Pedestrian Dead-Reckoning (PDR) to achieve more reliable and robust indoor positioning. Therefore, this section documents the state of the art with respect to the PDR, besides summarizing some existing indoor positioning methods which are related to the methods proposed in this work.

Pedestrian Dead-Reckoning

Generally, there are three core components in the PDR algorithm: step detection, step/stride length estimation and heading determination. The most common way in detecting a step is by observing peaks in acceleration signals. Some works have considered readings from gyroscopes and magnetometers in detecting steps [33]. The gait cycle of a human is composed of two phases: *Stance* and *Swing*. *Stance* phase is when the foot flattens firmly on the ground while *Swing* phase is when the foot swings forward to enter its *Stance* phase. The heel-touching-ground event and heel-off-ground event that occur during transitions between the two phases can cause spikes in the vertical acceleration, and so a step can be identified by recognizing local minimum and/or local maximum within specifically sized sliding windows [14, 81]. Regardless of the placement of the sensor/device, random jitters or movements other than walking can contribute to unwanted acceleration peaks and hence the false steps. To validate the detected step, thresholds have been introduced to be compared against various parameters such as the variance/standard deviation of accelerations within the sliding window [23, 57, 71], the magnitude difference between local maximum and local minimum [31, 35, 77], and the period between two consecutively detected steps [23, 41]. As the gait cycles during normal walking are fairly consistent, similar waveforms can be observed from sensor data. By measuring the similarities among the waveforms using techniques like Dynamic Time Warping (DTW) and Auto-correlation, the occurrence of step can thus be inferred [24, 44, 61]. For more robust detection of steps, classifiers are developed to distinguish the pedestrian's motion modes like walking, standing still and irregular movements based on a set of features extracted from sensor data [23, 73].

Once a step is identified, its length is important to indicate the distance traveled. The actual length of a step varies from person to person depending on factors like leg's length and walking speed. Persons that have longer legs tend to have bigger step length. Besides, bigger step length is also usually observed from higher walking speed. To walk faster, naturally the person increases the frequency of their gait cycle, and while doing so some body parts like hip and foot may experience sharper changes in acceleration. Therefore, models that relate the step length with frequency [23, 44, 57] or acceleration [31–33, 36, 79] have been

attempted for online estimation of step length for different walking patterns. Nonetheless, if the pedestrian walks normally at constant pace, implementing fixed step length for all the detected steps throughout the walk is deemed a viable, not to mention computationally much simpler, alternative because the variance of each step under such condition seems to be negligible [8, 71]. Fixed step length can be predetermined either through trials [15, 49, 71] or based on individual height and weight [14, 77, 81].

Aside from the inaccurately estimated distance traveled, erroneous heading has been reported as the dominant source of positioning errors in PDR based systems [1, 23, 31, 33, 41, 49, 71]. Basically, the heading is derived based on readings from sensors i.e. gyroscopes, accelerometers and magnetometers. Single integration of gyroscopes' readings gives angular displacement of the device while the digital compass composed of either individual magnetometers or fusion of accelerometers and magnetometers tells the device's absolute orientation i.e. azimuth. The device's orientation is not necessarily the pedestrian's heading as it is dependent on its placement on pedestrian e.g. in pocket or in swinging hand. If the device's orientation is aligned with pedestrian's heading, azimuth values are typically used to denote the pedestrian's headings [14, 23, 29, 33, 35, 81]. Otherwise, the very beginning heading of the pedestrian and/or device's orientation must be known beforehand so that the angular displacements computed using gyroscope or compass can be useful to deduce later headings [31, 41]. Either way, the estimated headings are bound to have errors because the compass is susceptible to unpredictable magnetic disturbances at times and the gyroscope' accuracy drifts over time [1]. The coupling of both compass and gyroscope however, as compared to each individual sensor, is able to achieve more reliable heading estimation [15, 23, 32, 40, 44, 49, 77, 83]. The common idea is to compare both compass and gyroscope data as an effort to mitigate the magnetic fluctuations or gyroscope's inherent biases, and hence fuse the data accordingly using techniques like Kalman filtering [32, 83], Particle filtering [44] and complementary filtering [15, 40, 49].

One popular means used in mitigating the sensor's inherent bias as well as the drift that results from integration of biased sensor data over time is called Zero-Velocity-Update (ZUPT) method. ZUPT method asserts that the foot is stationary at its *Stance* phase whereby its acceleration and velocity are supposedly zero. So, any non-zero accelerations and velocities that exist during *Stance* phase are considered errors and thus be reset to zero. Reliable detection of *Stance* phase from pedestrian's gait cycle is crucial to the effectiveness of ZUPT method. Therefore, the relevant sensors are usually mounted on the foot [2, 19, 33, 48, 83] because the movement of foot, as compared to that of other body parts, gives more apparent hints of gait cycles while walking.

Various schemes have also been attempted to complement PDR by making use of external or additional information. Some works like [41, 44, 57, 61] assume that the indoor map or floor plan is available and so information like constraints and paths can be retrieved from the map to rectify the estimated trajectory and heading. The corrections of estimated trajectory and heading in [77] are triggered upon detecting a virtual landmark. Virtual landmarks are identified by observing distinct signatures/patterns in the data extracted from magnetic and inertial sensors as well as Wi-Fi signals – for instance, overhearing a unique set of Wi-Fi access points at a particular corridor-turn. The actual location of each virtual landmark, however, is derived by repetitive estimation via PDR expecting the outcome would eventually converge to its actual location. In [23], besides having aids from Global Positioning System (GPS), map matching is triggered upon detection of certain patterns in user's movement based on accelerometers' reading. E.g. if the detected pattern is recognized as "using an elevator", the nearest elevator in vicinity is assumed as the user's position. In [36], the pedestrian is assumed carrying at least two mobile devices and the relative displacements of the devices with respect to the center of motion are reasonably stable as he/she walks. The devices individually estimate their own positions which is eventually corrected by a customized Maximum A Posteriori estimation. RSSI-based distance measurements derived from perceived wireless signals such as Ultrasound and Wi-Fi are used to complement PDR in [15, 35, 49]. Certainly, the use of Bayesian filter like Kalman filter (KF) [69] and Particle filter (PF) [5] is not uncommon among the existing works in fusing the available data and/or estimates for more optimal estimation. In general, better positioning performance is assured when more information is taken into consideration.

Collaborative Positioning

Another emerging means to gather more information for better position estimation is by utilizing the collaboration among the pedestrians to share the information among themselves. Collaborative positioning method is made possible due to the existence of clusters of pedestrians (who carry smart-devices) which is common in crowded places like mega-malls, universities and airports. The information shared among the pedestrians for collaborative positioning/navigation purposes can be of any types e.g. their relative/absolute position estimates and sensors' readings.

In [11], the FP-estimated position of a pedestrian is first corrected to a point where the resultant of forces exerted by nearby pedestrians is met. The force exerted by each nearby pedestrian corresponds to the confidence score which is computed based on the difference between its position estimated by FP and that of a PF. The corrected estimate is then fed into another PF to finalize the pedestrian's estimated position.

In [51], the pedestrians are expected to provide helpful feedback regarding their FP-estimated positions, which shall be used to update the database's credibility, thereby improving the subsequent FP estimates.

In [56], a group of pedestrians is assumed moving towards the same destination. Each pedestrian is provided an RSSI map which serves to deduce a set of likely positions based on real-time RSSI data. As the group moves, the set of likely positions shall be filtered according to their PDR-estimated trajectories, which eventually contains only a single estimate that concludes the group's estimated position.

In [1], the magnetometer readings from multiple pedestrians that are deemed reliable by a machine learning algorithm are fused by using consensus algorithms for more refined PDR-estimated headings. However, the pedestrians are assumed moving in the same direction, and at least one of them must have uncorrupted magnetometer readings.

In [29], the crowd of pedestrian is first categorized into groups based on two factors: firstly, RSSI-based proximity among the pedestrians and secondly, similarity between pedestrians' PDR-estimated trajectories. The PDR-estimated trajectory of each pedestrian shall be adjusted to reasonably fit the average PDR-estimated trajectory of their group. The estimated position of each pedestrian in the group is then re-adjusted according to the resultant of forces caused by nearby pedestrians as well as the groups.

In [72], the differences between pedestrians' PDR-estimated positions are constrained by a KF where the ranges derived from UWB signals are inputted as observations.

In [81], all the pedestrians share the same indoor map which has virtual grids on it. The pedestrians exchange information among themselves, which happens only when the distance between them is short enough for their microphone to sense frequency peaks broadcast from nearby pedestrians. The information received is the location of the center of grid where nearby pedestrian's PDR-estimated position lies in. The received locations are then averaged and fused with PDR-estimated position via a KF to finalize the estimated position of pedestrian.

In [28], the pedestrian's position is estimated using Ultrasound based Multi-Lateration where the references are *beacon* nodes and static nearby pedestrians.

In [31], PDR-estimated trajectory of the pedestrian is represented by a link structure where the links symbolize straight paths made by the pedestrian and the joints' angles denote the pedestrian's headings. When a pedestrian detects any nearby pedestrians, the joints' angles of the involved pedestrians are re-estimated in a least-squares sense.

In [74], the FP-estimated positions of pedestrians are adjusted using a spring model where estimated positions are represented as nodes and links/springs between the nodes have spring constants determined based on distance between nearby pedestrians as well as confidence

score of the FP estimates. The nodes get attracted or repulsed by the spring forces until equilibrium state is achieved.

2.5 Summary

This chapter gives an overview of the fundamentals of indoor positioning by discussing the major types of geometric measurements as well as the conventional positioning techniques. Besides, Kalman Filter and Particle Filter which are commonly used to fuse multiple types of measurements/information for better estimates are briefly described. In addition, state of the art regarding PDR is documented. Lastly, some existing works that relate to the proposed methods of this work are also summarized.

Chapter 3

Experimental Setup and Preliminaries

This chapter discusses about the preparation that is necessary prior to the evaluation of proposed positioning methods which shall be presented in the following chapters.

3.1 Setup

Two different sites in Swinburne University of Technology Sarawak Campus as depicted in Figure 3.1, were used to evaluate the positioning performance of the proposed methods. Figure 3.2 and Figure 3.3 show partial area of site no.1 and site no.2 respectively. During the experiments, the subject carried a Nexus 7 and walked along specific routes. All the necessary data were retrieved from the Nexus 7 and then post-processed using a *Matlab* program to compute the subject's positions.

Hardware

The hardware involved are listed as follows:

1. Asus Google Nexus 7 (version 2013)
2. "TP-Link Wireless N" Access Point (Model No.:TL-WA701ND) or Router (Model No.:TL-WR740N)

Their relevant specifications are shown in Table 3.1 and Table 3.2 respectively.

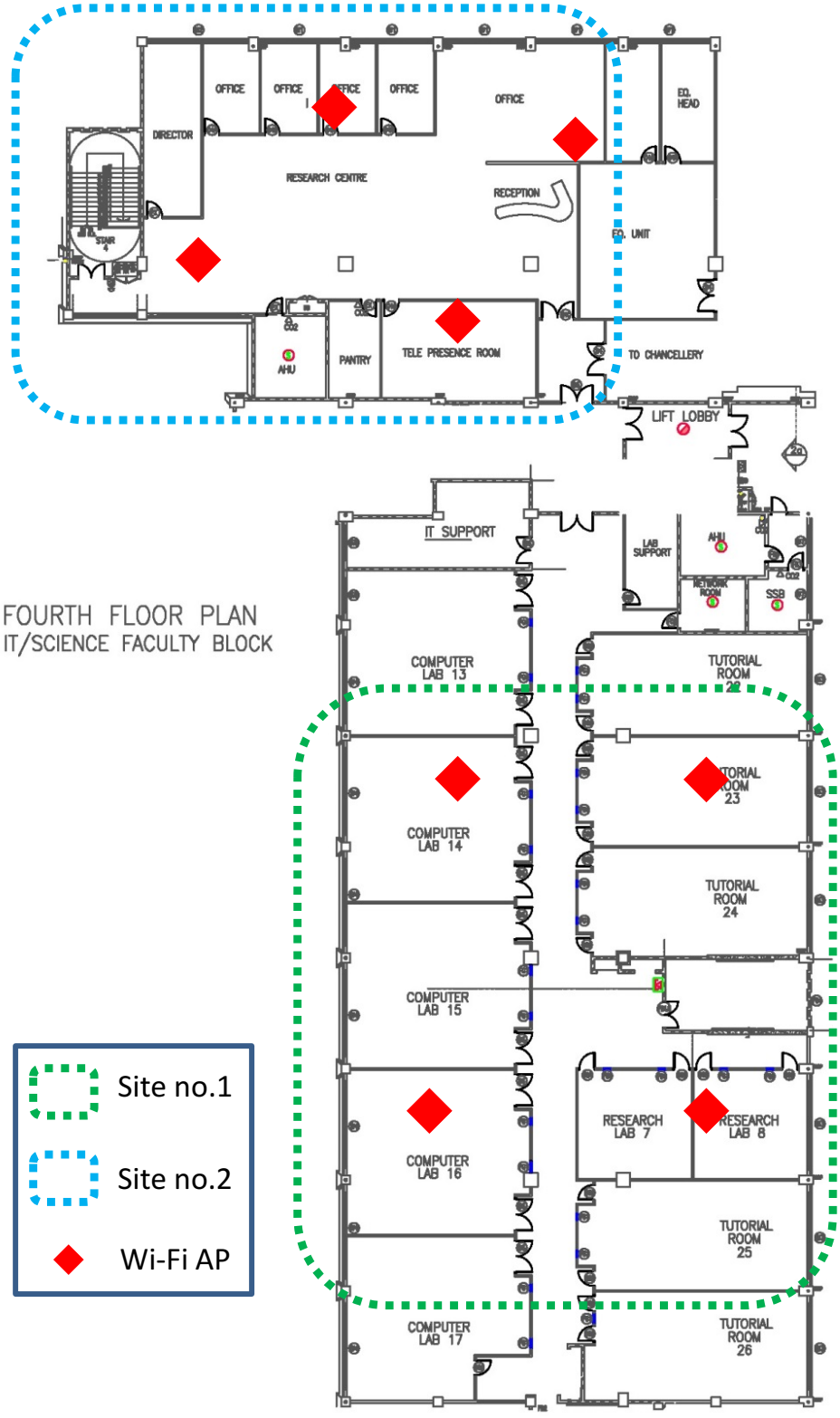


Fig. 3.1 Plane view of test-sites

Fig. 3.2 Long corridor of site no.1.

Fig. 3.3 Open-plan office of site no.2.

Table 3.1 Asus Google Nexus 7's specifications

Operating System	Android™ 4.3
CPU	Quad-core 1.5 GHz
Memory	2GB RAM
WLAN	Wi-Fi 802.11 a/b/g/n, dual-band
Sensor	Accelerometer, gyro, proximity, compass

Table 3.2 Specifications of TP-Link Router/Access Point

Standards	IEEE 802.11 b/g/n
Signal rates	Up to 150Mbps
Frequency range	2.4 – 2.4835 GHz
Antenna type	5dBi Omni-directional antenna
Transmit power	<20dBm (EIRP)
Beacon interval	100ms

Data Collection

An Android app was developed and run on the Nexus 7 in order to retrieve the necessary data (i.e. 3-axial linear accelerations, 3-axial angular rates, azimuth readings and RSSI values). The android app was written in Java using a software called *Android Developer Tools* following the Application Program Interface (API) guides which are available from the official website of *Android Developers*. All types of data were captured at a sampling rate of 50 Hz. Note that the actual refresh rate for RSSI measurements was approximately 1000ms.

Post-processing

The collected data were then inputted to a program written in *Matlab*. All types of collected data, except the azimuths, were filtered prior to the computation of subject's positions. Specifically, a second-order low-pass Butterworth filter with 0.1 Hz cut-off frequency was applied onto RSSI data. The accelerometers' readings were first processed via a moving average with a window size of 25, and subsequently a second-order low-pass Butterworth filter with a cutoff frequency of 5 Hz. The gyroscopes' readings were first processed via a moving average with a window size of 25, and subsequently a second-order low-pass Butterworth filter with a cutoff frequency of 1 Hz.

3.2 RSSI-Distance Correlation

There are two types of wireless signals considered in this work: Wi-Fi and Bluetooth. For either types of signals, the correlation between RSSI and distance can be expressed by Equation (2.6) whose parameters' values are shown in Table 3.3. The employed path-loss model's parameters' values for Wi-Fi signal were approximated by finding the best-fit curve, as depicted in Figure 3.4 where each shaped spot denotes the average value of RSSI readings acquired over a period of ten seconds from specific AP at random location within the site. The employed path-loss model's parameters' values for Bluetooth signal were adopted from [74]. Nonetheless, it is worth pointing out that derived values may be empirically adjusted.

Table 3.3 Employed path-loss model's parameters' values

Signal type	n	β
Wi-Fi	2.56	37.52
Bluetooth	2.85	40.35

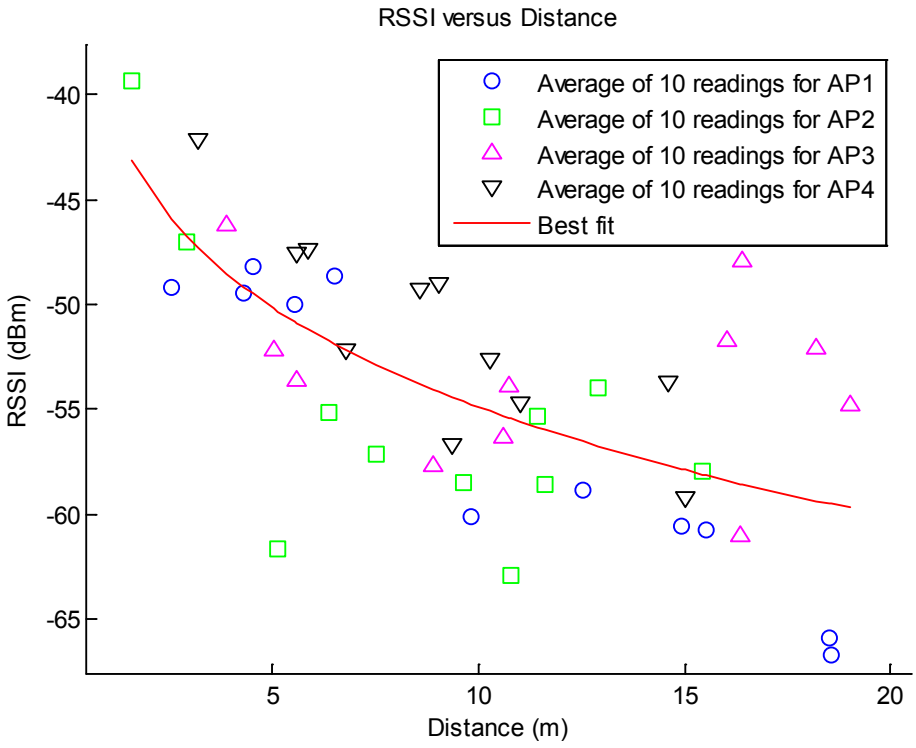


Fig. 3.4 RSSI-Distance correlation at site no.2

From Figure 3.4, one can observe that the average RSSI values are considerably inconsistent even at similar distances. This is inline with the common fact that RSSI readings

are erratic, and hence the reliability of the path-loss models are limited as they assume that the RSSI varies logarithmically and isotropically from the transmitter. Other than signal interference and NLOS conditions, the hardware itself is also one of the major factors that contribute to erratic RSSI values. The hardware may vary in terms of its chip-set as well as antenna type. Besides, the RSSI readings are also influenced by antenna's orientation [3, 20, 21, 75]. Nonetheless, the erratic RSSI data can be mitigated and made useful to a certain extent by empirically tuning the parameters of path-loss models or using appropriate fusion techniques along with some assumptions, as attempted in many existing works such as [15, 31, 42, 49, 52, 61, 74]. More sophisticated modelling of RSSI-distance correlation can be found in [52]. Note that in this work, the distances estimated based on RSSI values via a path-loss model are simply rough estimates which shall then be utilized along with custom filters and techniques to contribute better positioning accuracy.

3.3 PDR Implementation

Conventional PDR is employed to estimate the pedestrian's position on a per step basic by using Equation (2.7). The pedestrian is assumed to be walking normally with the device held either upright in hand or at waist level [15, 18, 64]. The assumption made is justifiable because it is intended in this work to demonstrate that, even under such optimistically controlled condition, PDR suffers primarily from erroneous headings. Needless to say, worse can be expected if PDR were to work under challenging conditions (e.g. irregular walking patterns and device placed in swinging hand/bag).

Step Detection

The length of each step is assumed fixed as the variance between each step's length is very small when a person walks normally [71]. To detect the walking steps of the pedestrian, firstly, the norm of accelerations is computed from the raw 3-axis linear accelerations obtained from the smart-device:

$$\tilde{S}^{a}_{rms}[t] = \sqrt{\tilde{a}^2_x[t] + \tilde{a}^2_y[t] + \tilde{a}^2_z[t]} \tag{3.1}$$

$\tilde{S}^{a}_{rms}$ is then processed by using a moving average filter with a window size of 25 to remove the waveform's mean value because only the variation in acceleration is needed. Subsequently a second-order low-pass Butterworth filter with a cutoff-frequency of 5 Hz is applied to remove the high-frequency noise, thereby producing smoother and clearer waveform to ease the identification of the peaks in the waveform. A step event is identified when there is a local maximum followed by a local minimum in the waveform. Note that threshold is necessary in identifying the local extrema among the computed norms of accelerations, and it is defined as 0.1 in this work. It is possible that the change in pedestrian's heading may cause change in the magnitude of accelerations and hence result in undesired signal peaks, in other words, false step events. To avoid false step events, 3-axis angular velocities are used to compute the norm of angular velocities first,

$$\tilde{S}^{w}_{rms}[t] = \sqrt{\tilde{w}^2_x[t] + \tilde{w}^2_y[t] + \tilde{w}^2_z[t]} \tag{3.2}$$

then it is processed via a moving average filter with a window size of 25, and subsequently a second-order low-pass Butterworth filter with a cutoff-frequency of 1 Hz. Then, any instance of $\tilde{S}^{w}_{rms}$ that falls within a certain range of magnitude is removed to further eliminate the unwanted noise so that the turn event can be easily identified. The turn event is recognized by detecting a full cycle of sinusoidal-like waveform. So, any step event that occurs during the turn event will be ignored. An example of this phenomenon is illustrated in Figure 3.5.

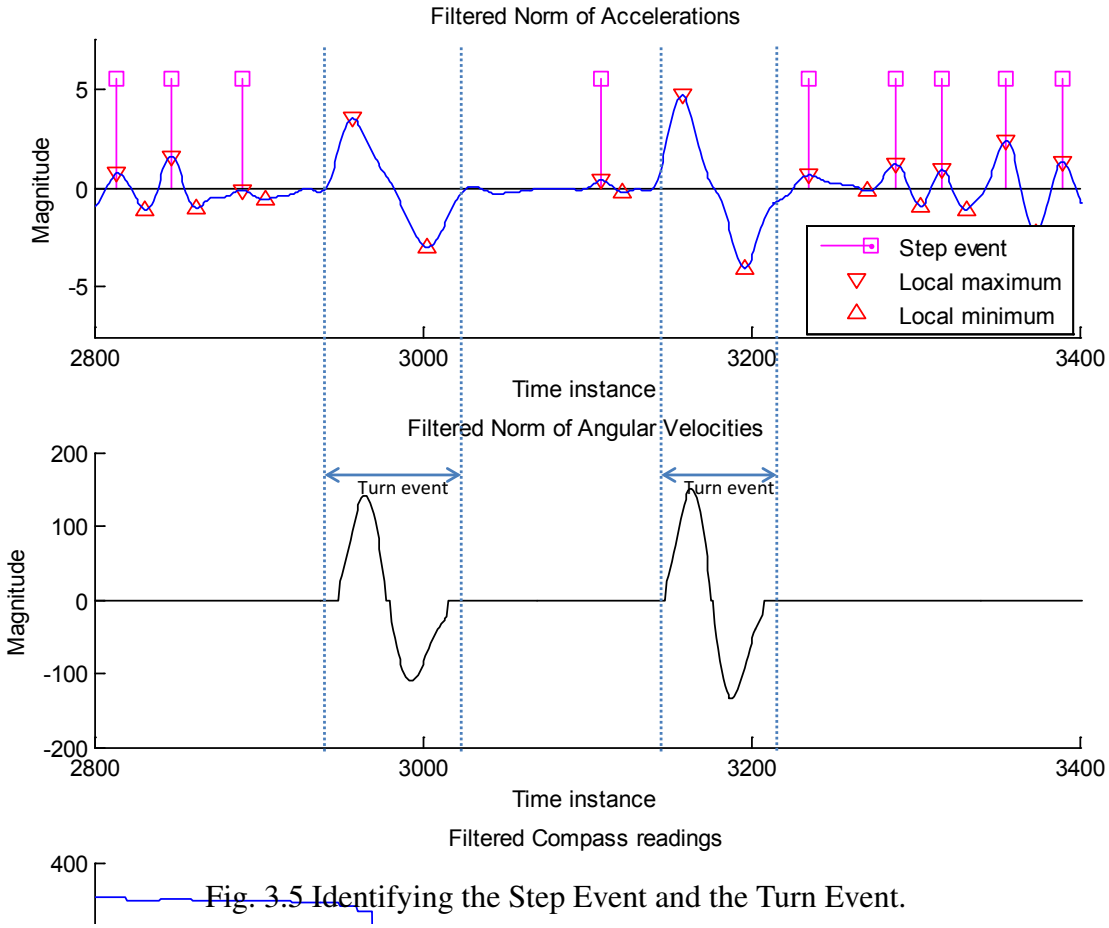


Fig. 3.5 Identifying the Step Event and the Turn Event.

Heading Estimation

The heading of the pedestrian ϑ_t can be obtained in two ways. The first way requires the pedestrian to hold the smart-device upright in hand so that the device's orientation is inline with pedestrian's actual heading, and hence their heading is simply computed by:

$$\vartheta_t = \theta \tag{3.3}$$

where θ denotes the azimuth value which is extracted directly from smart-device via Android's Application Program Interface (API). In contrast, the second way of acquiring $\vartheta s,t$ is applied when the device's orientation is unknown, and hence pedestrian's heading is estimated by:

$$\vartheta_t = \vartheta_{t-1} + \Delta\theta \tag{3.4}$$

where $\Delta\theta$ is the change in azimuth during the turn event. However, it assumes that the device is attached at the waist level of pedestrian and the initial heading (ϑ_0) is known. Figure 3.6 shows an example on acquisition of $\Delta\theta$ from the obtained azimuth values. Note that the

waveforms shown in Figure 3.6 are resulted from firstly a 220 degree anticlockwise rotation and subsequently a 120 degree clockwise rotation. According to the API guides provided by official website of *Android Developers*, the azimuth values are derived from fusions of readings of accelerometers and magnetometers. As accelerometers output bias will drift according to ambient temperature and magnetometers may experience magnetic disturbances due to ferromagnetic materials inside building, the derived azimuth values are bound to have

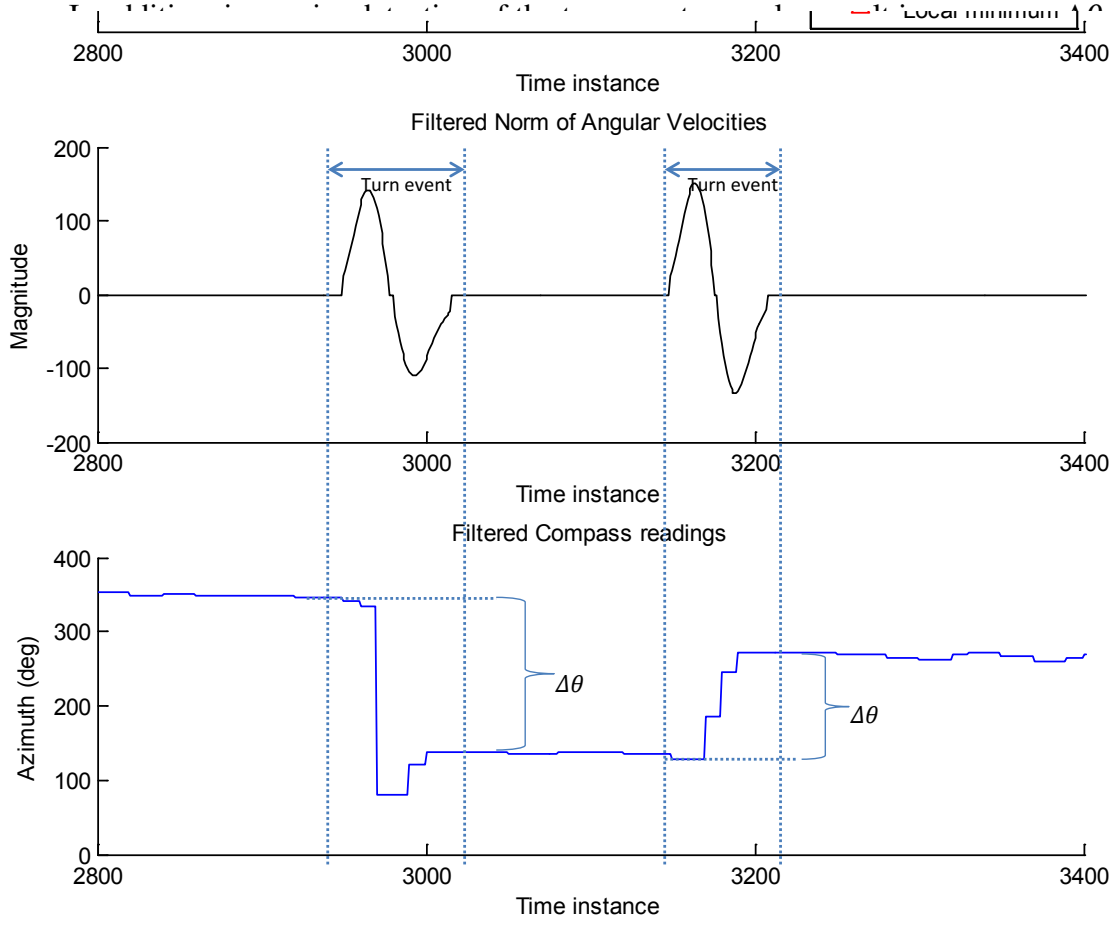


Fig. 3.6 Computing the change in Azimuth.

3.4 Benchmarks

A number of some other positioning methods were prepared as benchmarks for the methods proposed in this work, and they are listed as follows:

1. ***aTRI***: RSSI -based Tri-Lateration
2. ***pSIR***: Fusion of PDR and RSSI via Sampling Importance Resampling Particle Filter
3. ***pMCMC***: Fusion of PDR and RSSI via Markov Chain Monte Carlo sampling
4. ***pKF***: Fusion of PDR and Tri-Lateration via Kalman Filter

The selection of positioning methods for a fair comparison in terms of positioning performance is rather difficult as the existing methods are uniquely designed to work optimally with specific requirements. For instance, the Fingerprinting or landmarks based methods can be highly reliable only when the number of references is high which would require laborious and time-consuming site-surveys while some exploit external information like nearby pedestrians, indoor map and GPS. In short, better positioning performance comes with greater requirement in several aspects like available information, hardware and preparation work. The four existing methods listed above are chosen as benchmarks mainly because they can be executed by using the same hardware involved in the proposed methods.

3.4.1 Algorithm of *aTRI*

Adapted from [53], ***aTRI*** estimates the position of pedestrian by using RSSI-derived distances from three Wi-Fi APs whose locations are assumed known. In cases where more than three APs are available, only APs whose RSSI values are top three highest among all are considered.

An iteration of ***aTRI*** algorithm is summarized by **Algorithm 1** where t denotes the time instance; P is estimated position of pedestrian; and d_a, d_b, d_c are the distances estimated via a path-loss model based on the RSSI values obtained from three individual APs whose 2-dimensional Cartesian coordinates are represented by (x_a, y_a), (x_b, y_b) and (x_c, y_c) respectively.

Algorithm 1 An iteration of *aTRI* algorithm

1: $A \leftarrow \left((d_{b,t})^2 - (d_{c,t})^2\right) - \left((x_b)^2 - (x_c)^2\right) - \left((y_b)^2 - (y_c)^2\right)$
2: $B \leftarrow \left((d_{b,t})^2 - (d_{a,t})^2\right) - \left((x_b)^2 - (x_a)^2\right) - \left((y_b)^2 - (y_a)^2\right)$
3: Ynum $\leftarrow B \cdot (x_c - x_b) - A \cdot (x_a - x_b)$
4: Yden $\leftarrow (y_a - y_b)(x_c - x_b) - (y_c - y_b)(x_a - x_b)$
5: Define Y-coordinate: $Y \leftarrow$ Ynum $\div$ Yden
6: Xnum $\leftarrow A - Y \cdot (y_c - y_b)$
7: Xden $\leftarrow x_c - x_b$
8: Define X-coordinate: $X \leftarrow$ Xnum $\div$ Xden
9: $P_t \leftarrow \begin{bmatrix} X & Y \end{bmatrix}^\intercal$

3.4.2 Algorithm of *pSIR*

pSIR uses a Sampling Importance Resampling (SIR) Particle Filter to combine PDR with RSSI-derived distance measurement. The use of SIR Particle Fitler in ***pSIR*** is very similar to that of [35]. However, ***pSIR*** derives the distance via a path-loss model by using the RSSI data, while [35] measures the distance based on propagation time of Ultrasound signals. The latter way of distance measurement is comparatively more reliable, but it can only be performed when the Ultrasound emitter is less than few meters apart from the pedestrian; on the other hand, Wi-Fi signal can reach far longer distances, but ***pSIR*** utilizes only RSSI values that pass the RSSI threshold which is to be set empirically. Besides, unlike [35] that considers only single distance at a time, ***pSIR*** may consider multiple distances at a time if multiple RSSI sources are seen. Having said that, this work simply assumes the positioning performance of ***pSIR*** is equivalent to that of [35].

The pedestrian's position is estimated on a per step basis. An iteration of ***pSIR*** algorithm is summarized by **Algorithm 2** where t denotes the time instance; P equates $\begin{bmatrix} X & Y \end{bmatrix}^\intercal$ where X and Y are the x- and y-coordinates of pedestrian; N is the sample size; S denotes a set of samples; L is the step length; ϑ is the PDR-estimated heading; ε_x and ε_y are random noises which are zero-mean Gaussian; σ_r is the standard deviation of measurement noise; D denotes a set of distances between pedestrian and Wi-Fi APs whose RSSI values pass the desired threshold; and λ is the distance between the k-th sample and the n-th AP. Initial position (P_0) is assumed known, and initial set of samples (S_0) is generated from a Gaussian distribution ($\mathcal{N}(P_0, \sigma)$).

The procedure of **Algorithm 2** can be summarized as follows:

1. (Lines no.1 to 10): A set of samples is generated to represent the likely positions of pedestrian. The weight of each generated sample is then assigned based on D.

2. (Lines no.11 to 14): All samples' weights are normalized.

3. (Lines no.15 to 28): A new set of samples is drawn from the previously generated set of samples, with probabilities proportional to their normalized weights.

4. (Line no.29): The mean of the new set of samples finalizes the estimated position of pedestrian.

Algorithm 2 An iteration of *pSIR* algorithm

for $k \leftarrow 1, N$ **do**
    Generate a sample: $S_t^{[k]} \leftarrow S_{t-1}^{[k]} + L \times \left[\cos\vartheta_t \;\; \sin\vartheta_t\right]^\intercal + \left[\varepsilon_x \;\; \varepsilon_y\right]^\intercal$
    Compute sample's weight, $w_t^{[k]}$:
    $E \leftarrow$ number of elements in D
    if $E > 0$ **then**
        $w_t^{[k]} \leftarrow \prod_{n=1}^{E} \left(\frac{1}{\sqrt{2\pi}\sigma_r} \exp\left(-\frac{1}{2(\sigma_r)^2}(D^{[n]} - \lambda)^2\right)\right)$
    else
        $w_t^{[k]} \leftarrow 1$
    end if
end for
Compute total weight: $W \leftarrow \sum_{n=1}^{N} w_t^{[n]}$
for $k \leftarrow 1, N$ **do**
    Normalize sample's weight: $w_t^{[k]} \leftarrow w_t^{[k]} \div W$
end for
Initialize cumulative distribution function (CDF): $c^{[1]} \leftarrow 0$
for $k \leftarrow 2, N$ **do**
    Construct CDF: $c^{[k]} = c^{[k-1]} + w^{[k]}$
end for
Start at the bottom of CDF: $m \leftarrow 1$
Draw a starting point: $u^{[k]} \leftarrow\sim \mathcal{N}(0, \frac{1}{N})$
for $k \leftarrow 1, N_s$ **do**
    Move along CDF: $u^{[k]} = u^k + \frac{1}{N}(k-1)$
    while $u_k > c_m$ **do**
        $m \leftarrow m + 1$
    end while
    Assign sample: $S_t^{[k]} \leftarrow S_t^{[m]}$
    Assign weight: $w_t^{[k]} = \frac{1}{N}$
end for
$P_t \leftarrow \frac{1}{N} \sum_{l=1}^{N} S_t^{[l]}$

3.4.3 Algorithm of *pMCMC*

pMCMC employs Markov Chain Monte Carlo (MCMC) sampling in fusing PDR with RSSI-derived distances, as adapted from [49]. In [49], it is reported that their MCMC-based fusion scheme outperforms that of [34] which is based on Maximum Likelihood estimation.

The pedestrian's position is estimated on a per step basis. An iteration of ***pMCMC*** algorithm is summarized by **Algorithm 3** where t denotes the time instance; P equates $\begin{bmatrix} X & Y \end{bmatrix}^\intercal$ where X and Y are the x- and y-coordinates of pedestrian; L is the step length; ϑ is the PDR-estimated heading; ε_x and ε_y are random noises which are zero-mean Gaussian; N is the sample size; E is the number of elements in D; D denotes a set of distances between pedestrian and Wi-Fi APs whose RSSI values pass the desired threshold; σ_r is the standard deviation of measurement noise; λ_A is the distance between proposed sample (s) and n-th AP; λ_B is the distance between current sample ($S^{[k-1]}$) and n-th AP.

Algorithm 3 An iteration of *pMCMC* algorithm

Initialize first sample: $S^{[1]} \leftarrow P_{t-1} + L \times \begin{bmatrix} \cos\vartheta_t & \sin\vartheta_t \end{bmatrix}^\intercal + \begin{bmatrix} \varepsilon_x & \varepsilon_y \end{bmatrix}^\intercal$
for $k \leftarrow 2, N$ **do**
  Propose a sample: $s \leftarrow S^{[k-1]} + \begin{bmatrix} \varepsilon_x & \varepsilon_y \end{bmatrix}^\intercal$
  Compute likelihood ratio, K:
  if ($E \leftarrow$ number of elements in D) > 0 **then**
    $A \leftarrow \prod_{n=1}^{E} \left(\frac{1}{\sqrt{2\pi}\sigma_r} \exp\left[-\frac{1}{2(\sigma_r)^2} (D^{[n]} - \lambda_A)^2 \right] \right)$
    $B \leftarrow \prod_{n=1}^{E} \left(\frac{1}{\sqrt{2\pi}\sigma_r} \exp\left[-\frac{1}{2(\sigma_r)^2} (D^{[n]} - \lambda_B)^2 \right] \right)$
    $K \leftarrow A \div B$
  else
    $K \leftarrow 0$
  end if
  Define acceptance ratio: $\alpha \leftarrow$ minimum value between K and 1.0
  if $\alpha \geq$ desired threshold **then**
    Accept proposed sample: $S^{[k]} \leftarrow s$
  else
    Reject proposed sample: $S^{[k]} \leftarrow S^{[k-1]}$
  end if
end for
Compute mean of samples: $P_t \leftarrow \frac{1}{N} \sum_{l=1}^{N} S^{[l]}$

The procedure of **Algorithm 3** can be summarized as follows:

1. The first element of the sample set (which comprises N number of elements) is initialized based on a desired model, as detailed by line no.1.

2. A new sample is proposed by introducing random noise into the previous sample, as detailed by line no.3.

3. The new sample is accepted as the next element of the sample set, only if the acceptance ratio is not smaller than certain threshold. Otherwise, the next element shall be the same as the previous element. The acceptance ratio is computed as detailed by lines no.4 to 12.

4. Once all the elements of the sample set are computed, the mean of the sample set finalizes the estimated position of pedestrian, as detailed by line no.19.

3.4.4 Algorithm of *pKF*

Kalman Filter [69] has been employed to combine PDR with additional information like nearby pedestrians' locations (in [81]) and GPS readings (in [23]). In this work, none of those additional information but the RSSI data is available. Therefore, the RSSI data are used to compute the pedestrian's position via *aTRI* whose estimate is then inputted as measurement in the Kalman Filter to finalize the estimated position of pedestrian.

The pedestrian's position is estimated on a per step basis. An iteration of ***pKF*** algorithm is summarized by **Algorithm 4** where t denotes the time instance; P equates $\begin{bmatrix} X & Y \end{bmatrix}^\intercal$ where X and Y are the x- and y-coordinates of pedestrian; L is the step length; ϑ denotes the PDR-estimated heading; A, B and H are 2×2 identity matrices; both Q and R are 2×2 identity matrices multiplied by some appropriate scalars; z is pedestrian's position estimated simultaneously by ***aTRI***. The initial position of pedestrian (P_0) is assumed known, and initial error covariance ($\mathbf{P}_0$) equates Q. Note that lines no.5 to 9 in **Algorithm 4** are standard equations for Kalman Filtering, as described in **Chapter 2.3**.

Algorithm 4 An iteration of ***pKF*** algorithm

1: Define control input: $u \leftarrow L \times \begin{bmatrix} \cos\vartheta_t & \sin\vartheta_t \end{bmatrix}^\intercal$
2: **if** Pedestrian makes a turn **then**
3: Reset error covariance: $\mathbf{P}_t \leftarrow Q$
4: **end if**
5: Predict next state: $x \leftarrow A \times P_{t-1} + B \times u$
6: Predict next covariance: $\mathbf{P}_t \leftarrow A \times \mathbf{P}_{t-1} \times A^\intercal + Q$
7: Compute Kalman gain: $K \leftarrow \mathbf{P}_t \times H^\intercal \left(H \times \mathbf{P}_t \times H^\intercal + R\right)$
8: Update predicted state: $x \leftarrow x + K \times (z_t - H \times x)$
9: Update predicted covariance: $\mathbf{P}_t \leftarrow \mathbf{P}_t - K \times H \times \mathbf{P}_t$
10: $P_t \leftarrow x$

The procedure of **Algorithm 4** can be summarized as follows:

1. The pedestrian's next position (i.e. x) is first predicted by by projecting forward the previously estimated position (i.e. P_{t-1}) with known displacements, as detailed by line no.5. The next covariance (i.e. $\mathbf{P}_t$) is predicted by adding the current covariance ($\mathbf{P}_{t-1}$) with some random noise (i.e. Q), as detailed by line no.6.

2. The Kalman gain (i.e. K) is computed based on some random noise (i.e. R), as detailed by line no.7. The Kalman gain shall be used to update both the predicted position and predicted covariance, as detailed by lines no.8 to 9.

3. The updated x finalizes the estimated position of pedestrian.

3.5 Summary

Two different sites within the Swinburne University of Technology Sarawak Campus were used for the evaluations of the positioning methods proposed in this work. The hardware required were Nexus 7 tablets and Wi-Fi routers/access points. All the necessary data were retrieved from the Nexus 7 via an Android app and then post processed by a *Matlab* program to estimate the subject's locations. The path-loss model's parameters' values were defined by best fitting the pre-collected RSSI data at known distances. Besides, the algorithms of PDR as well as some other PDR-based positioning methods adapted from existing works are also explained in this chapter.

Chapter 4

Indoor Positioning by Pedestrian Dead-Reckoning with RSSI-based schemes

As mentioned in **Chapter 2.4**, many efforts have already been made to enhance the positioning performance of PDR by employing different techniques in processing the data from PDR-related sensors data (i.e. accelerometers, gyroscopes and magnetometers), as well as by making use of additional but external information such as indoor map, GPS readings and nearby pedestrians' data. In view of widespread Wi-Fi routers or access points in indoor environments, the RSSI of Wi-Fi signals has been favored by many researchers in developing indoor positioning methods especially those which are based on Fingerprinting. The Fingerprinting based methods are undoubtedly very promising, yet their relatively high setup requirements are unfavorable.

This chapter presents two unique schemes that aim to aid smart-device -based PDR by utilizing the RSSI of Wi-Fi signals. The first scheme (detailed in **Chapter 4.1**) adjusts the PDR-estimated positions of pedestrian by detecting strong RSSI and subsequently requiring pedestrian's manual input. Admittedly, the first scheme is not viable in situations where user intervention is not allowed. The second scheme (detailed in **Chapter 4.2**) rectifies the PDR's dominant error source, i.e. erroneous heading, upon detecting long straight trajectory in pedestrian's motion.

4.1 Position Correction Scheme

The procedure of position correction scheme is briefly shown in Figure 4.1. Conventional PDR algorithm (detailed in **Chapter 3.3**) is employed to provide initial estimates for pedestrian's positions, assuming the pedestrian holds the smart-device upright in hand. The correction on pedestrian's estimated position is first executed upon observing strong RSSI from Wi-Fi APs. Alternatively, further correction may be triggered with manual input from the pedestrian. Such trigger is deemed reasonable in situations where pedestrian needs to react accordingly to the system for further navigation guidance.

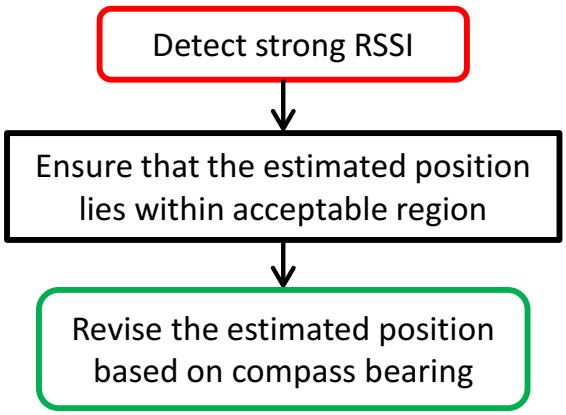


Fig. 4.1 Flowchart of Position Correction Scheme.

4.1.1 Methodology

As the pedestrian walks, the RSSI readings obtained by the smart-device (held by pedestrian) from each visible APs are filtered and recorded as $R_{t,i}$, where $i \in \{1,2,\cdots,N_A\}$, t denotes the time instance and N_A is the total number of visible APs. In the course of time, $R_{t,i}$ is checked against certain RSSI threshold $R_{th,i}$. The $R_{th,i}$ corresponds to certain level of proximity (represented by $d_{th,i}$) that exists between the estimated position P_t and i^{th} AP. If $\left(R_{t,i} \geq R_{th,i}\right)$ is met, then $\left(d_{t,i} \leq d_{th,i}\right)$ is expected. The $d_{t,i}$ is the distance between the estimated position P_t and i^{th} AP, and it is computed by

$$d_{t,i} = \sqrt{(X_t - X_i)^2 + (Y_t - Y_i)^2} \tag{4.1}$$

where X_i and Y_i are the Cartesian coordinates of i^{th} AP.

So, if $\left(R_{t,i} \geq R_{th,i}\right)$ and $\left(d_{t,i} \leq d_{th,i}\right)$, then this P_t is within the "acceptable region" and no further action is required. However, if $\left(R_{t,i} \geq R_{th,i}\right)$ but $\left(d_{t,i} \geq d_{th,i}\right)$, then this P_t is not accepted and shall be shifted radially towards the AP and subsequently relocated to a point where $\left(d_{t,i} \leq d_{th,i}\right)$ is fulfilled. Such phenomenon can be illustrated by an example depicted in Figure 4.2. Multiple acceptable regions can be assigned if the RSSI-distance relationship is reliable over a longer distance, and would lead to more effective correction. Note that the RSSI value in reality does not vary isotropically from the RSSI source; regardless, this scheme simply assumes the opposite.

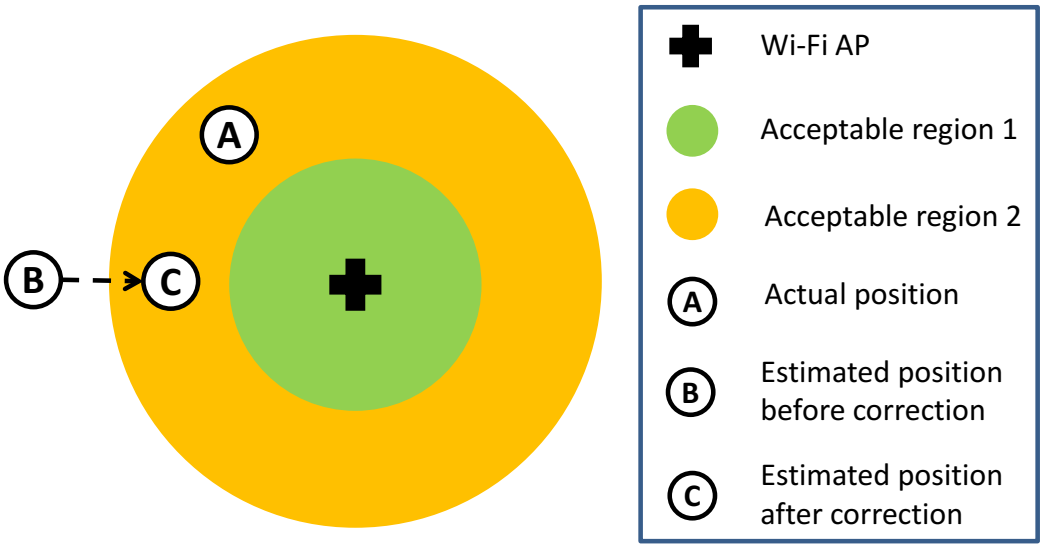


Fig. 4.2 Position 'B' is not accepted and is therefore being shifted radially to Position 'C'.

The estimated position can be further improved by utilizing the compass bearing (denoted as α_t^i) measured from i^{th} AP to pedestrian. Whenever the condition $\left(R_{t,i} \geq R_{th,i}\right)$ is detected, the pedestrian is prompted to turn towards the corresponding AP and then trigger the system to capture α_t^i at that moment. Note that the APs should be placed in a way whereby they can be spotted easily by the pedestrian. Admittedly, such scheme is only applicable in a controlled environment, e.g. museum; APs could be placed on top of certain displayed objects in the museum, and so the visitors may approach an AP while viewing the displayed objects. The sole purpose of having α_t^i is to acquire vector displacement between pedestrian and AP. Subsequently, the pedestrian's estimated position is finalized according to the acquired vector displacement, as illustrated by an example depicted in Figure 4.3 assuming that the captured α_t^i is 315° clockwise from North.

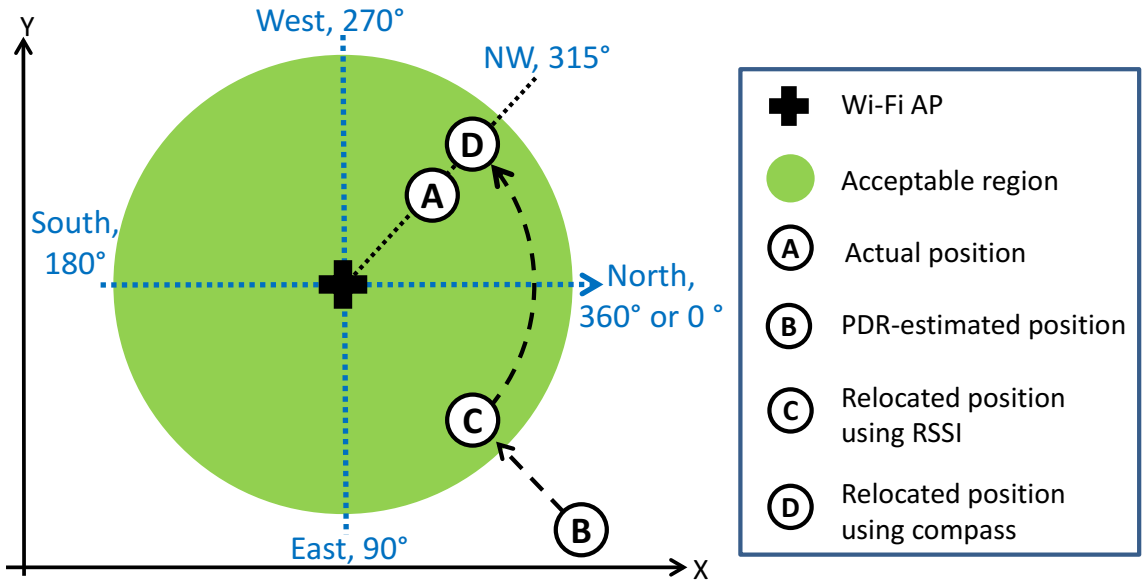


Fig. 4.3 Position 'C' is shifted to Position 'D' based on the acquired compass bearing.

4.1.2 Evaluation

The proposed method had been tested in an open-plan office. A total of ten trials were conducted. In each trial, the subject held a Nexus 7 upright in hand and walked along the route beginning from 'Start' until 'End' as illustrated in Figure 4.4. Note that though the condition $\left(R_{t,i} \geq R_{th,i}\right)$ may be fulfilled at any time along the route, the system was only allowed to check whether condition $\left(R_{t,i} \geq R_{th,i}\right)$ was achieved at checkpoints (i.e. 'C4', 'C8', 'C10' and 'C13'.) only and to trigger correction accordingly. The RSSI-distance correlation of the site was considerably poor, as described in Chapter 3.2. In the experiment, two RSSI thresholds were employed and defined as -45 dBm and -50 dBm that correspond to 2 m and 5 m respectively, thereby resulting in two acceptable regions as indicated by dotted circles (with different radii) shown in Figure 4.4. The RSSI thresholds were selected not to be smaller than -50 dBm because RSSI values lower than -50dBm were deemed relatively unreliable, besides avoiding the acceptable regions of an AP from overlapping those of another AP.

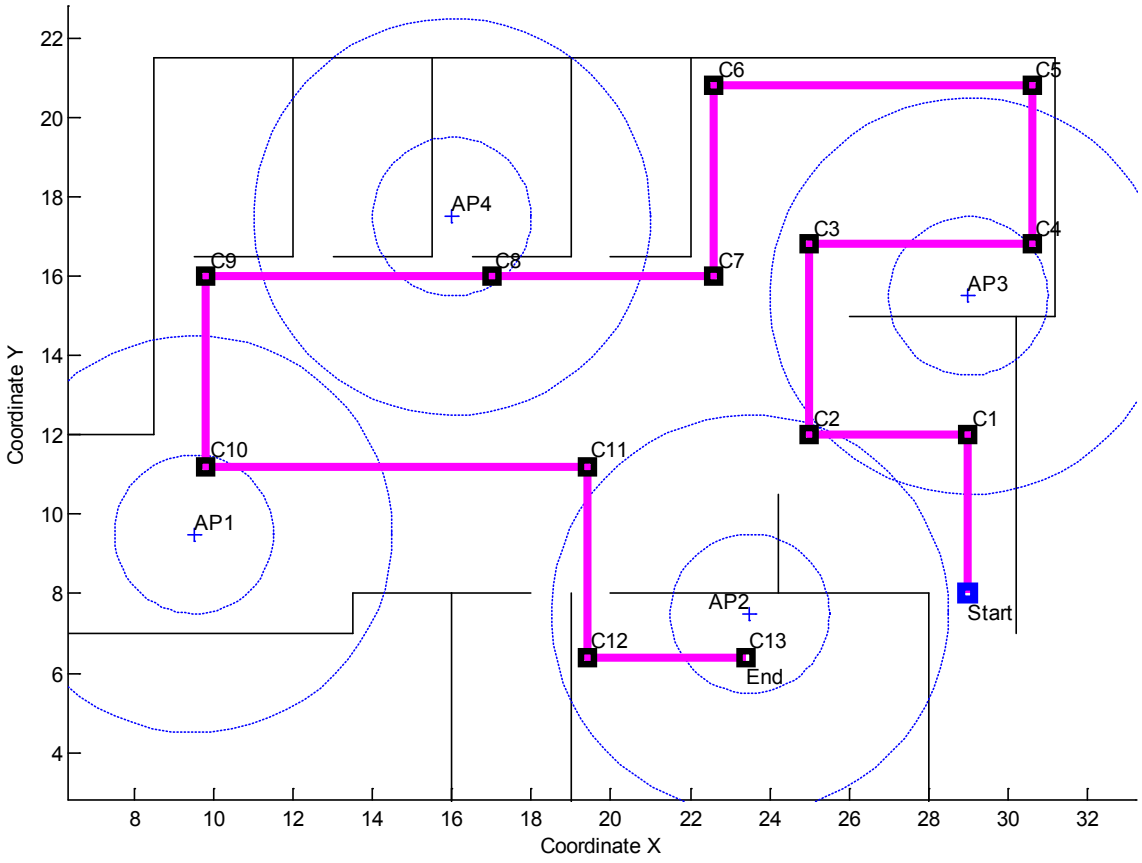


Fig. 4.4 Plane view of testbed.

The proposed scheme is similar to those of [49], [35] and [15], in a sense that the pedestrian's PDR-estimated position is refined upon being in vicinity of beacon nodes. Specifically, Markov Chain Monte Carlo (MCMC) sampling is used in [49] to fuse PDR estimates with distances derived from RSSI values which pass the threshold; Sampling Importance Resampling (SIR) Particle Filter (PF) is employed in [35] to combine PDR estimates with distances derived from nearby (not more than few meters away from) ultrasound device; in [15], the pedestrian's estimated position is directly assumed to be the position of beacon node whose RSSI value is the highest and passes certain threshold. For comparison purposes, a total of five positioning methods were considered in estimating the subject's positions, and they are listed as follows:

1. ***pPDR***: Sole PDR, as explained in **Chapter 3.6**.
2. ***pSIR***: PDR with Sampling Importance Resampling, as detailed in **Chapter 3.7.2**
3. ***pMCMC***: PDR with Markov Chain Monte Carlo sampling, as detailed in **Chapter 3.7.3**.
4. ***pBN***: Same as *pPDR*, but that the estimated position eventually gets replaced by the position of Beacon Node whose RSSI value passes the threshold.
5. ***PM***: Same as *pPDR*, but that the estimated position eventually gets adjusted by the proposed correction scheme.

Impact of varying number of deployed APs

The error metric is the absolute difference between estimated position and actual position. The lower the error, the higher the positioning accuracy. Figure 4.5 shows the errors of all the five different methods when the number of deploys APs varied according to the arrangements shown in Table 4.1. From Figure 4.5, it can be observed that mean error of *PM* gradually decreases as the number of APs deployed increases, with lowest mean error of 1.4 m and standard deviation of 1.8 m while *pPDR*'s are constantly 3.2 m and 1.8 m because PDR is not dependent on RSSI at all. Nonetheless, the improvement made on all methods except *pPDR* by increasing the number of deployed APs from 2 to 4 units seems insignificant because even the biggest difference between corresponding mean error is just 0.6 m. This is probably owing to the PDR component of *PM* that can estimate relatively accurate though for a short duration. The accumulative error of PDR will eventually become significant, and that is exactly the time where the pedestrian needs to be close to the beacon nodes (i.e. APs) so that strong RSSI values may be available for all these RSSI-dependent methods (i.e. *pMCMC*, *pSIR* , *pBN*, *PM*) to execute appropriate correction. For example, as shown in Figure 4.6, the estimated route beginning from the 'Start' deviates progressively from actual route due to mere PDR with miscounted steps and primarily erroneous headings; once greatly improved at checkpoint 'C8', the estimated route remains relatively close to the actual route all the way from 'C8' until the 'END' with little improvement made at checkpoints 'C12' and 'C13'; therefore, in this case, 'AP2' is deemed not as necessary as 'AP4'. Nevertheless, as more APs are deployed along the route, there are more chances that the correction can be triggered when necessary. Note that the estimated positions depicted in Fig. 4.6 are not mean values, but computed on a per step basis; when a step event occurs, the filtered RSSI value (which is not lower than the selected RSSI threshold) at that instant only shall be considered in updating the PDR-estimated position.

Table 4.1 Arrangements of Wi-Fi APs considered during experiment

Arrangement No.	Sources of RSSI considered
1	AP1, AP2, AP3, AP4
2	AP1, AP2, AP3
3	AP1, AP3, AP4
4	AP1, AP2, AP4
5	AP2, AP3, AP4
6	AP1, AP2
7	AP1, AP3
8	AP1, AP4
9	AP2, AP3
10	AP2, AP4
11	AP3, AP4

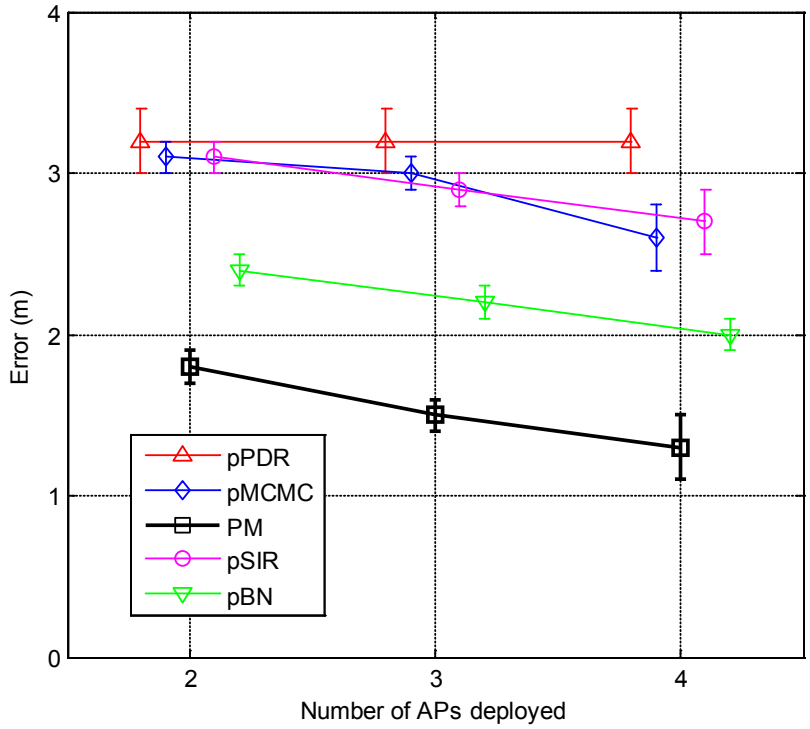


Fig. 4.5 Errors of different methods Versus Number of APs deployed

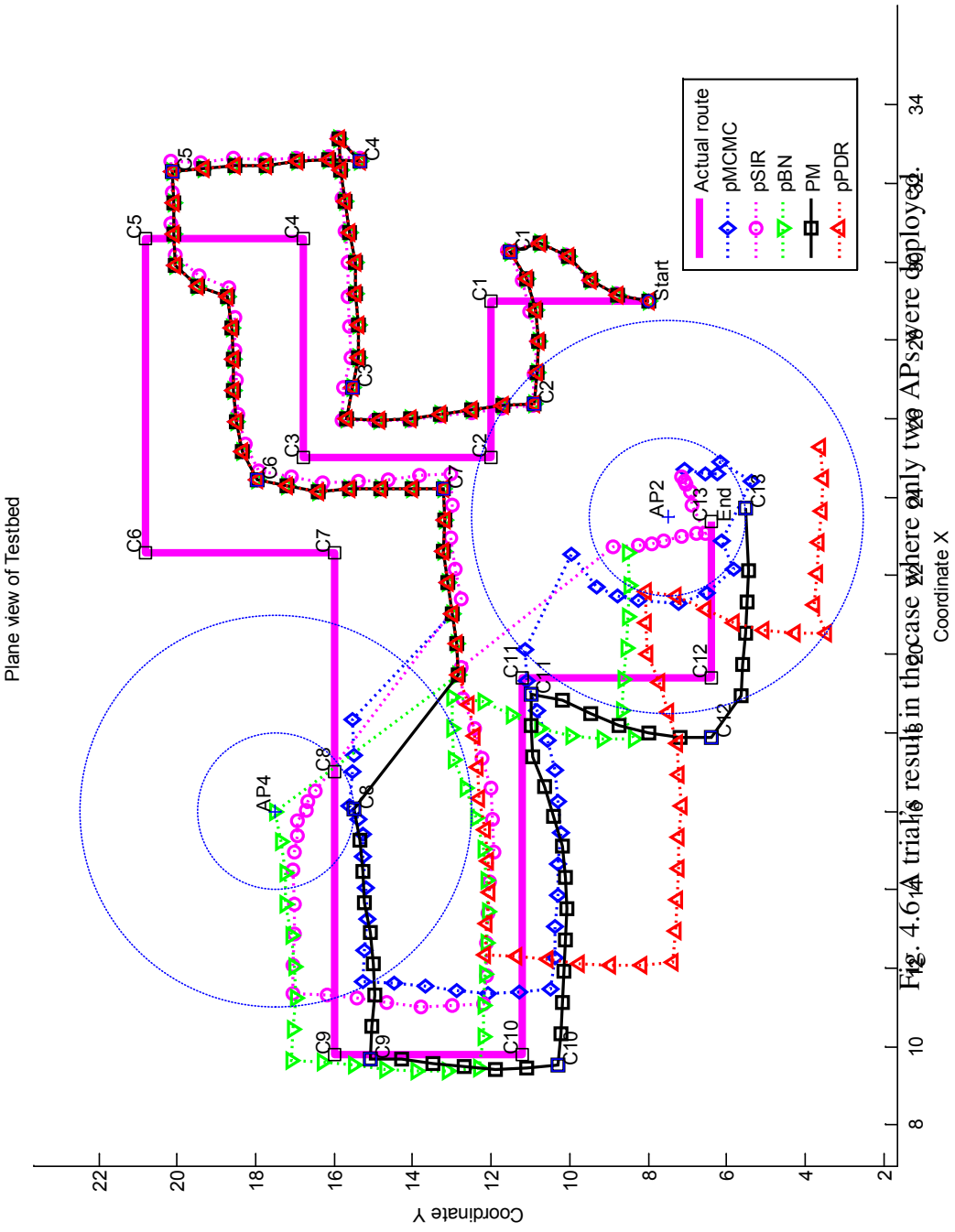


Fig. 4.6 A trial's results in the case where only two APs were deployed.

Impact of varying RSSI Threshold

The method *PM* had two RSSI thresholds set as -45 dBm and -50 dBm respectively while the remaining methods except *pPDR* had only one RSSI threshold set as -50 dBm. The RSSI thresholds have to be empirically defined for optimal estimation. If RSSI threshold is too high, correction is unlikely to happen unless the pedestrian is indeed near an AP to obtain sufficiently strong RSSI readings. If RSSI threshold is too low, correction gets triggered unnecessarily and tends to fail because weak RSSI values are rather poor in implying the distances between pedestrian and AP. Figure 4.7 and Figure 4.8 illustrate two examples of outcomes of faulty correction caused by erratic RSSI value or inappropriate RSSI threshold. In Figure 4.7, the condition $\left(R_{t,i} \geq R_{th,i}\right)$ was met and hence pedestrian turned towards 'AP3' and captured the α_t^i (the bearing of pedestrian from 'AP3') to execute the proposed correction scheme; however, as $R_{t,2}$ (RSSI from 'AP2') is greater than $R_{t,3}$ (RSSI from 'AP3') at that moment, *PM* mistakenly assumed the pedestrian was within the acceptable region of 'AP2' instead of 'AP3' and therefore finalized the estimated position to a point where the bearing of estimated position from 'AP2' fulfilled the captured α_t^i. In Figure 4.8, the $R_{t,2}$ (RSSI from 'AP2') obtained by the pedestrian at checkpoint 'C6', which was supposedly lower than the RSSI threshold, unexpectedly passed the RSSI threshold and therefore led to faulty *pBN*.

Basically, both *PM* and *pBN* expect that the pedestrian's closest AP gives highest RSSI value whenever the condition $\left(R_{t,i} \geq R_{th,i}\right)$ is met. In other words, the "acceptable region" of an AP is not allowed to overlap with that of another AP. In contrast to them, the methods *pMCMC* and *pSIR* theoretically can make use of RSSI readings from any APs at a time. However, from Figure 4.9 that depicts the errors of *pMCMC* and *pSIR* for different RSSI thresholds, it is apparent that the -50 dBm was the optimal RSSI threshold for both *pMCMC* and *pSIR* in such test-site. Nonetheless, their errors are still worse than those of *pBN* and *pPM*, as shown in Figure 4.5. This thus infers that both the methods *pMCMC* and *pSIR* in the experiment ended up needing rather strong RSSI values (not lesser than -50 dBm), just like *PM* and *pBN*, but yielding comparatively lower positioning accuracy. Such poor performance of both *pMCMC* and *pSIR* was probably caused by rather poor RSSI data, as their algorithms involve deriving the distances from RSSI values via certain path-loss model. From Figure 4.10 which shows the average absolute difference between RSSI-derived distances and actual distances at every checkpoint of the route, it is apparent that the RSSI data obtained during the experiment are far from being reliable in deriving the distances between pedestrian and APs. Note that RSSI-derived distances for all four APs were indeed computed via the same path-loss model.

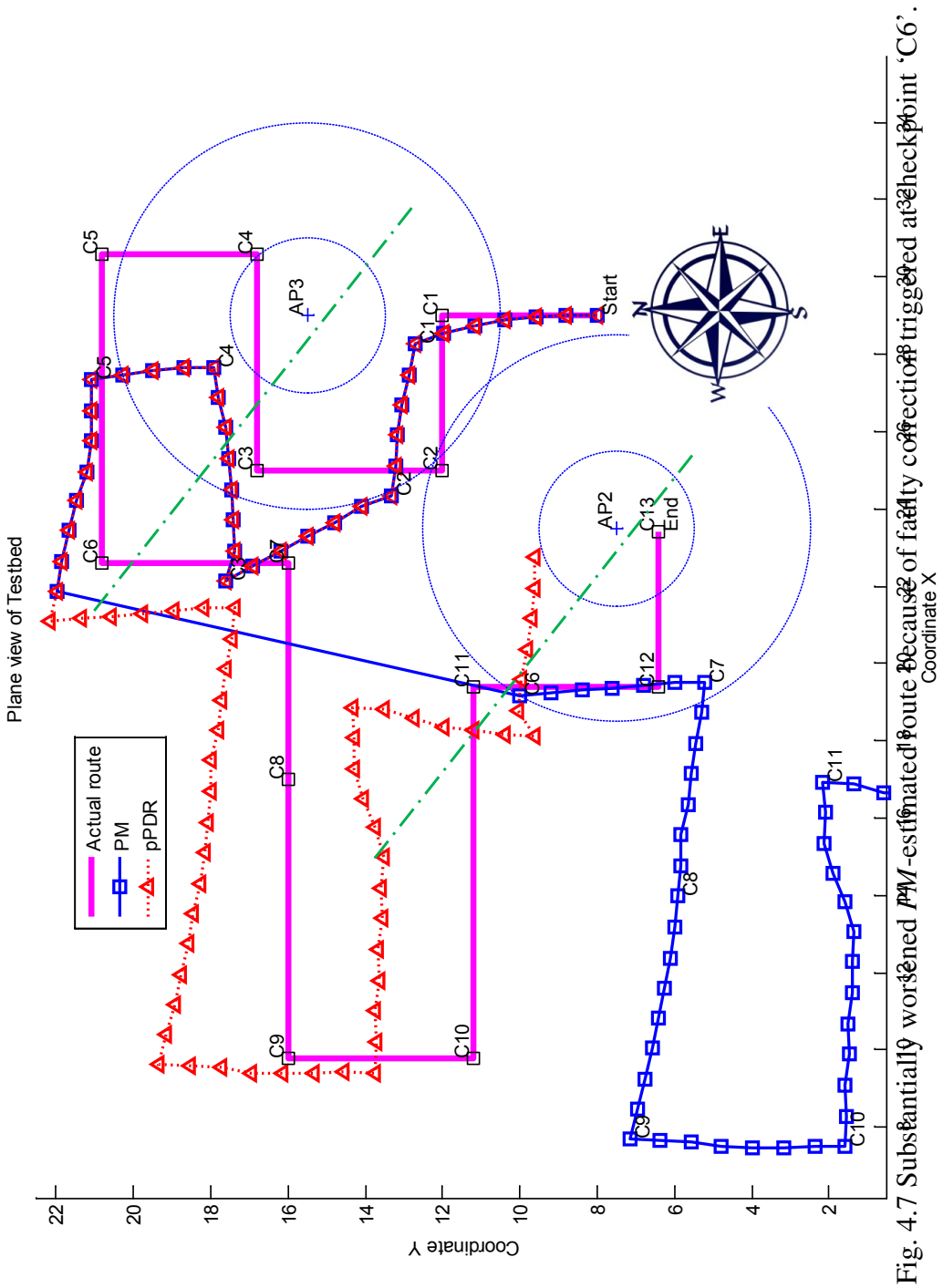


Fig. 4.7 Substantially worsened *PM*-estimated route because of faulty correction triggered at checkpoint 'C6'.

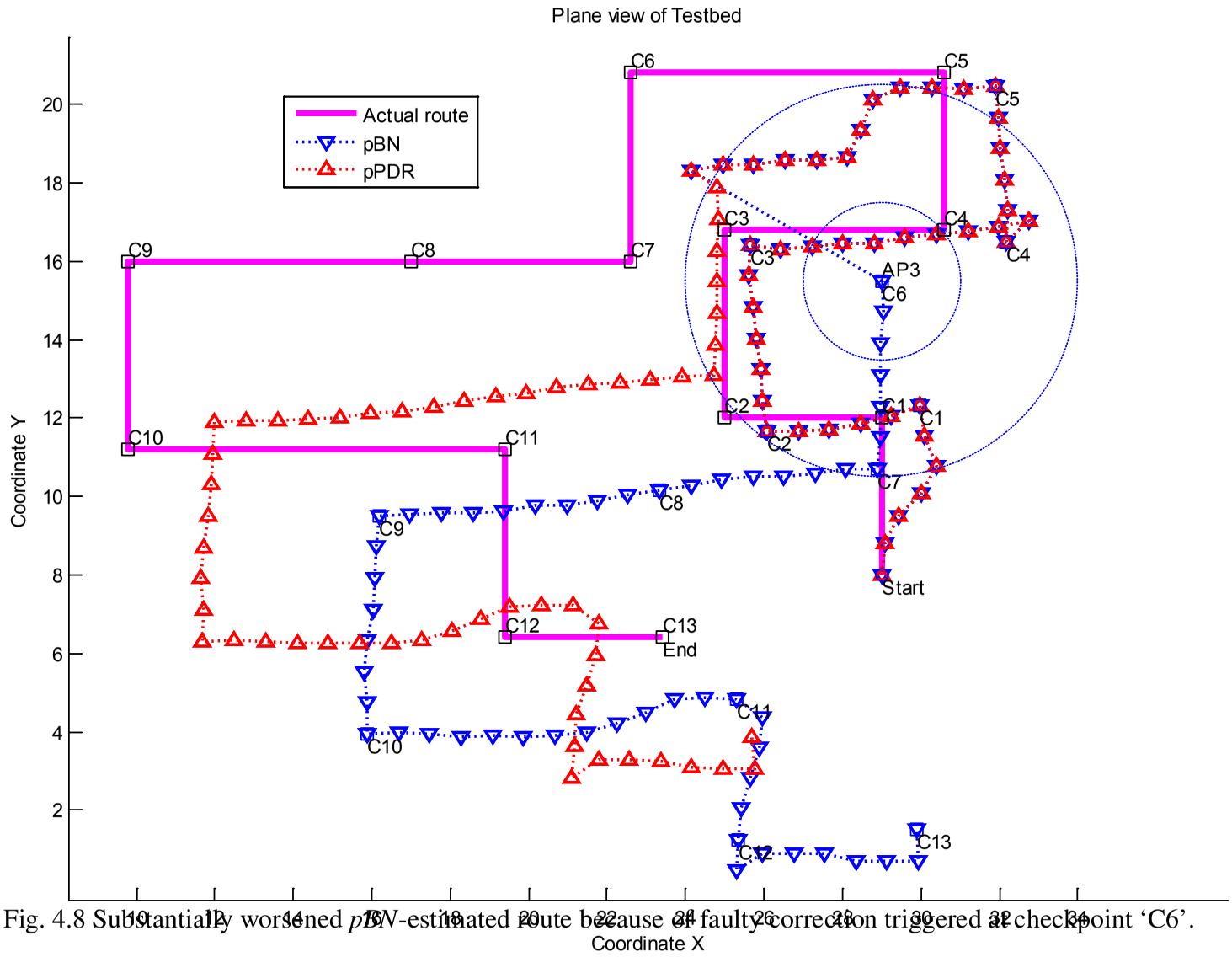


Fig. 4.8 Substantially worsened *pBN*-estimated route because of faulty correction triggered at checkpoint 'C6'.

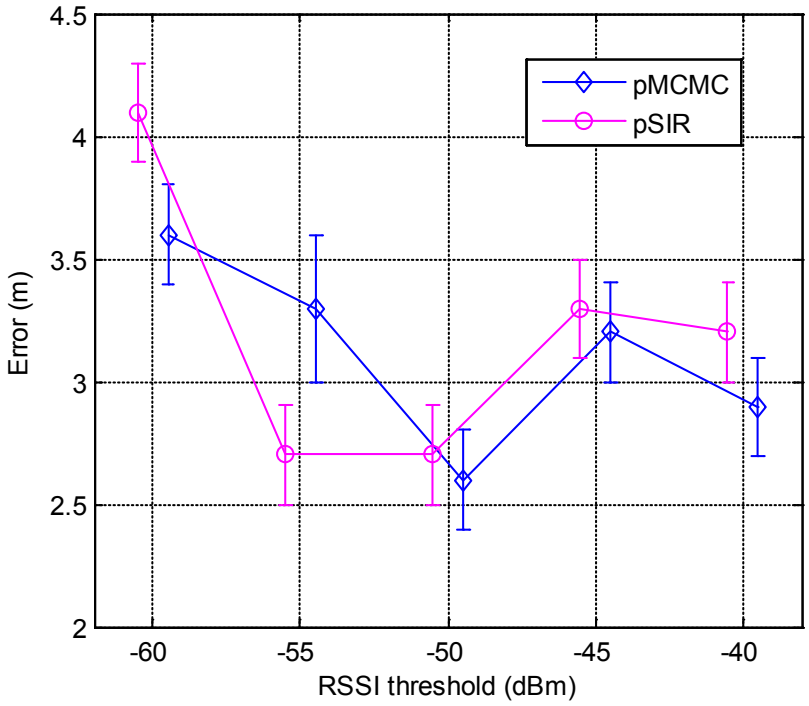


Fig. 4.9 Errors Versus RSSI threshold. (Case: Four APs deployed)

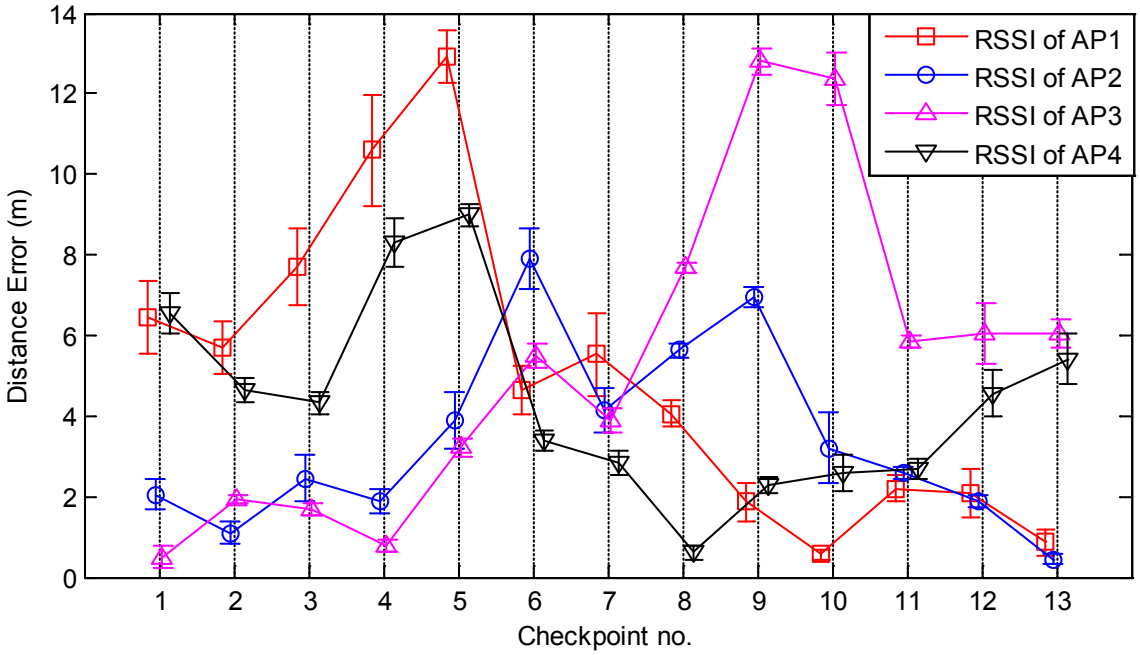


Fig. 4.10 Average absolute difference between RSSI-derived distances and actual distances at all checkpoints along the route.

Discussion

Admittedly, the fact that the pedestrian is required to manually trigger the system to capture the compass bearing (α_t^i) in order to acquire vector displacement for correction purpose may render the proposed scheme not viable in applications where user intervention is prohibited. This scheme is not practical as it inconveniences the user to disrupt normal movement along a desired walking path to aid error correction. Nevertheless, the scheme is for further improvement and not the core technique itself. *PM* without vector displacement just simply ensures that the estimated position lies within the acceptable region, and when its acceptable region is too small because of unreasonably high RSSI threshold, it becomes somewhat similar to *pBN* which outperformed both *pMCMC* and *PM* in the experiment anyway.

Regardless, this work intends to emphasize the importance of having vector displacement between pedestrian and AP, which is apparent especially in scenarios like the one shown in Figure 4.11. In Figure 4.11, the PDR-estimated route is way too much swayed from the actual route that it falls into the northern region of 'AP4' when approaching 'C8'; knowing only scalar displacements between pedestrian and 'AP4' can only shift the estimated positions towards/outwards 'AP4' radially, just like how both *pMCMC* and *pSIR* do, and therefore the resultant estimated positions still fall within the northern region of 'AP4'; *PM*, which knows the vector displacement between pedestrian and 'AP4', manages to shift the estimated position happening at 'C8' all the way from the North-East of 'AP4' to the South of 'AP4' and the resultant estimated positions remain relatively closer to their actual positions.

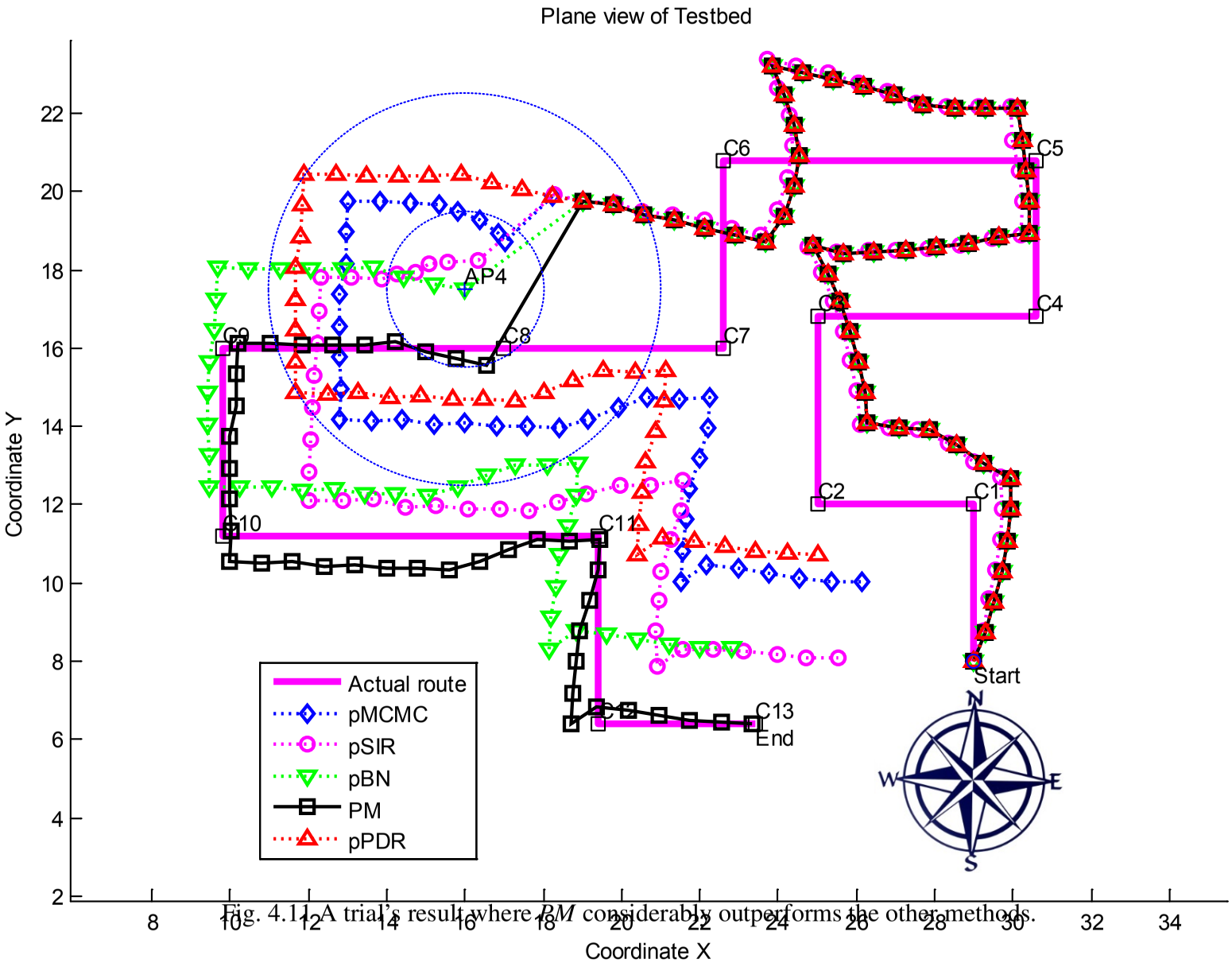


Fig. 4.11 A trial's result where *PM* considerably outperforms the other methods.

4.2 Heading Correction Scheme

This scheme aims to correct the PDR-estimated headings ϑ_t by making use of the RSSI data. Under this scheme, the pedestrian neither needs to approach any RSSI sources nor provides manual inputs. The correction of PDR-estimated headings, however, gets triggered only when the pedestrian's movement constitutes a long straight trajectory. Such trigger is deemed acceptable because long straight trajectory is commonly observed in pedestrian's daily movements, e.g. moving along long straight pathways in university, mega-mall, etc. The procedure of the heading correction scheme is briefly shown in Figure 4.12.

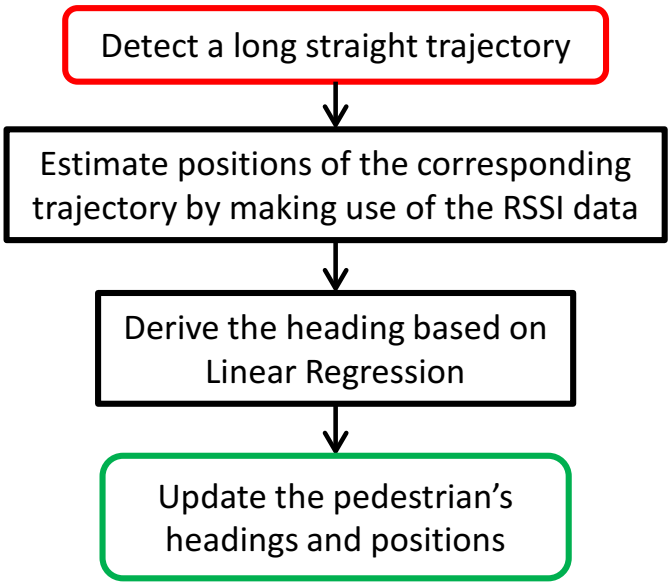


Fig. 4.12 Flowchart of Heading Correction Scheme.

4.2.1 Methodology

Detection of long straight trajectory

A straight trajectory can be identified by observing a series of sequential positions estimated by Equation (2.7), for instance $(P_t, P_{t+1}, P_{t+2}, \cdots)$ which occurs between the turn events. The size of the series implies the length of the straight trajectory. A straight trajectory is considered long only when its series' size passes the threshold η. It is undeniable that the pedestrian's trajectory across a long straight corridor/pathway may not be a completely straight line at times. For example, slight deviation of course is performed to avoid bumping into another person who is moving along the same route. Even so, the straight trajectories of the pedestrian make before and after the course's deviation can still be considered long if their corresponding series' size passes the threshold η. The threshold η has to be empirically defined so that the detected straight trajectory is sufficiently long enough to enable a more reliable derivation of heading. Note that the detected long straight trajectory is termed as *series* from now on.

Use of RSSI data for estimation of *series*' positions

The RSSI data R_i during the time-frame of *series* are used to estimate their corresponding d_i (which denotes the distance between pedestrian and i-th AP) based on a radio propagation model as follows:

$$R_i = -\beta - 10n\log_{10} d_i \tag{4.2}$$

where β and n are predefined constants to fit the site's RSSI-distance correlation [15, 74].

A Sampling Importance Resampling (SIR) Particle Filter (PF) is employed to fuse PDR estimates (i.e. steps and headings) with RSSI-derived distances d_i in estimating the positions of the entire *series*. Optimal estimation can be achieved by fine tuning the filter's parameters. Map information is commonly used in a PF to restrict the estimated positions to fall on pathways instead of walls or obstacles [44, 57, 61]. However, in this work, indoor map is considered not available and thus only the derived distances d_i are inputted as measurements to the particle filter.

Let $I_t = [\mathrm{x}_t, \mathrm{y}_t]^\intercal$ denote the position where x_t and y_t denote the coordinate x and y respectively at time t. Given the initial position I_{t-1}, N_s samples of I_t are generated using the proposed motion model:

$$I_t^k = I_{t-1}^k + S \times [\cos\vartheta_t, \sin\vartheta_t]^\intercal + [\varepsilon_x, \varepsilon_y]^\intercal \tag{4.3}$$

where $k = 1, \ldots, N_s$ and S is the step's length. Both ε_x and ε_y are $\sim \mathcal{N}(0, {\sigma_q}^2)$ where σ_q is the standard deviation of the process noise. The weight of each generated sample, w^k is computed by:

$$w^k = \begin{cases} \prod_{i=1}^{N_A} \frac{1}{\sqrt{2\pi}\sigma_r} \exp\left[-\frac{(d_i - g_i^k)^2}{2(\sigma_r)^2} \right] & \text{if } N_A \geq 1 \\ 1 & \text{if } N_A = 0 \end{cases} \tag{4.4}$$

where σ_r is the standard deviation of the measurement noise. g_i^k is the distance between sample I_t^k and i-th AP, computed by:

$$g_i^k = \| I_t^{[k]} - W_i \| \tag{4.5}$$

where W_i denotes the location of i-th Wi-Fi AP which is assumed known. Then, the weights are normalized:

$$w^k = \frac{w^k}{\sum_{l=1}^{N_s} w^l} \tag{4.6}$$

Finally, N_s samples are drawn from the generated set of samples $\{I_t^k\}_{k=1}^{N_s}$ with probabilities proportional to their w. With re-sampled set $\{I_t^k\}_{k=1}^{N_s}$, position I_t is simply the centroid of the samples [35, 44]:

$$I_t = \frac{1}{N_s} \sum_{k=1}^{N_s} I_t^k \tag{4.7}$$

Derivation of Heading based on Linear Regression

Once all the positions of the entire *series* (which is the detected long straight trajectory) are computed, a simple linear regression line is to be determined through these positions. Simple linear regression is a statistical method that models the relationships between a dependent variable and an explanatory variable. As each position of the *series* has two variables namely x_t and y_t, a right choice has to be made between regression of y_t on x_t and regression of x_t on y_t.

Consider a case where an object travels in a direction inclining to the horizontal axis rather than to the vertical axis. In such case, one can infer that the object's horizontal displacement is greater than its vertical displacement and hence regression of y_t on x_t is more appropriate to resemble the object's direction. Likewise deduction can be applied for the opposite case. As for the *series*, the displacement made can be represented by the difference between the *series*' first and last positions as all the positions of *series* are sequentially estimated by a

SIR PF. Let I_{first} and I_{last} denote the first and the last position of the *series* respectively. If $\|\mathrm{x}_{first} - \mathrm{x}_{last}\| \geq \|\mathrm{y}_{first} - \mathrm{y}_{last}\|$, then regression of y_t on x_t is chosen. Otherwise, regression of x_t on y_t is chosen. The orientation of the linear regression line is regarded as the new heading denoted as Θ. Admittedly, some estimated positions of the *series* could be outliers due to erroneous RSSI readings. In this work, the outliers are indiscriminately considered in deriving the new headings. The impact of the number of outliers on estimation accuracy shall be looked into in future work.

Finally, the pedestrian's headings ϑ during the time-frame of the *series* are replaced by Θ. All the corresponding previously estimated positions $P_{s,t}$ are re-estimated using (Equation (2.7)) with this newly updated ϑ. An example of such phenomenon is illustrated in Figure 4.13. Note that it is desirable to have longer *series* so that more estimated positions can be considered to reliably derive the linear regression line.

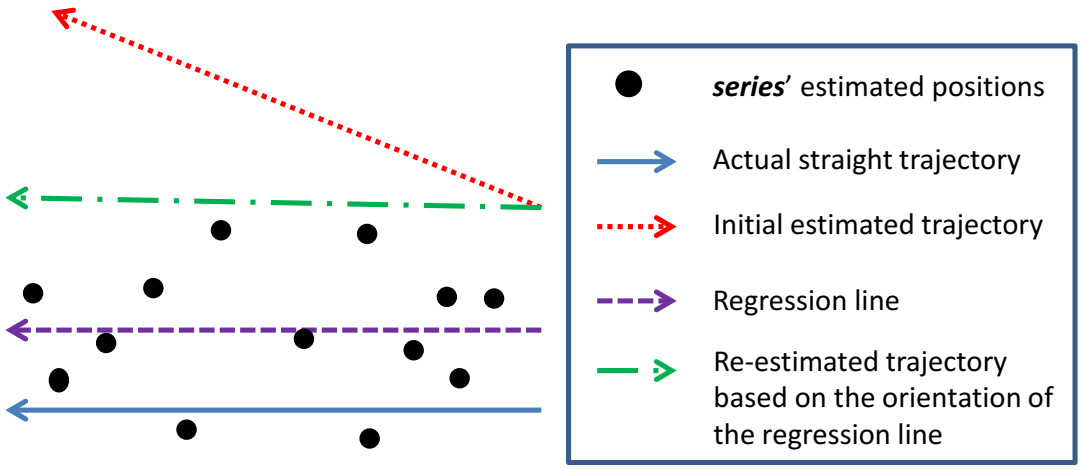


Fig. 4.13 The original headings of the PDR-estimated trajectory are updated to the orientation of the linear regression line. Hence, the estimated trajectory is re-estimated with updated headings.

4.2.2 Evaluation

Figure 4.14 and Figure 4.15 depict the plan views of the sites which were used to evaluate the efficacy of proposed heading correction scheme in improving the positioning accuracy of PDR. In terms of site's area and no. of *n*odes deployed, the sites chosen are comparable to those of [15, 35, 42, 76]. Though not shown, the sites are fully equipped with machines, furniture as well as walls or barriers made of both glass and concrete. Furthermore, the experiments were conducted during normal operating hours of the sites, which means a considerable number of students and relevant staff were present at the sites. All these obstacles and disturbances present at the sites undoubtedly contribute to erratic RSSI measurements. The purpose of conducting the experiments in such scenario is to test the robustness of the proposed method towards erratic RSSI values.

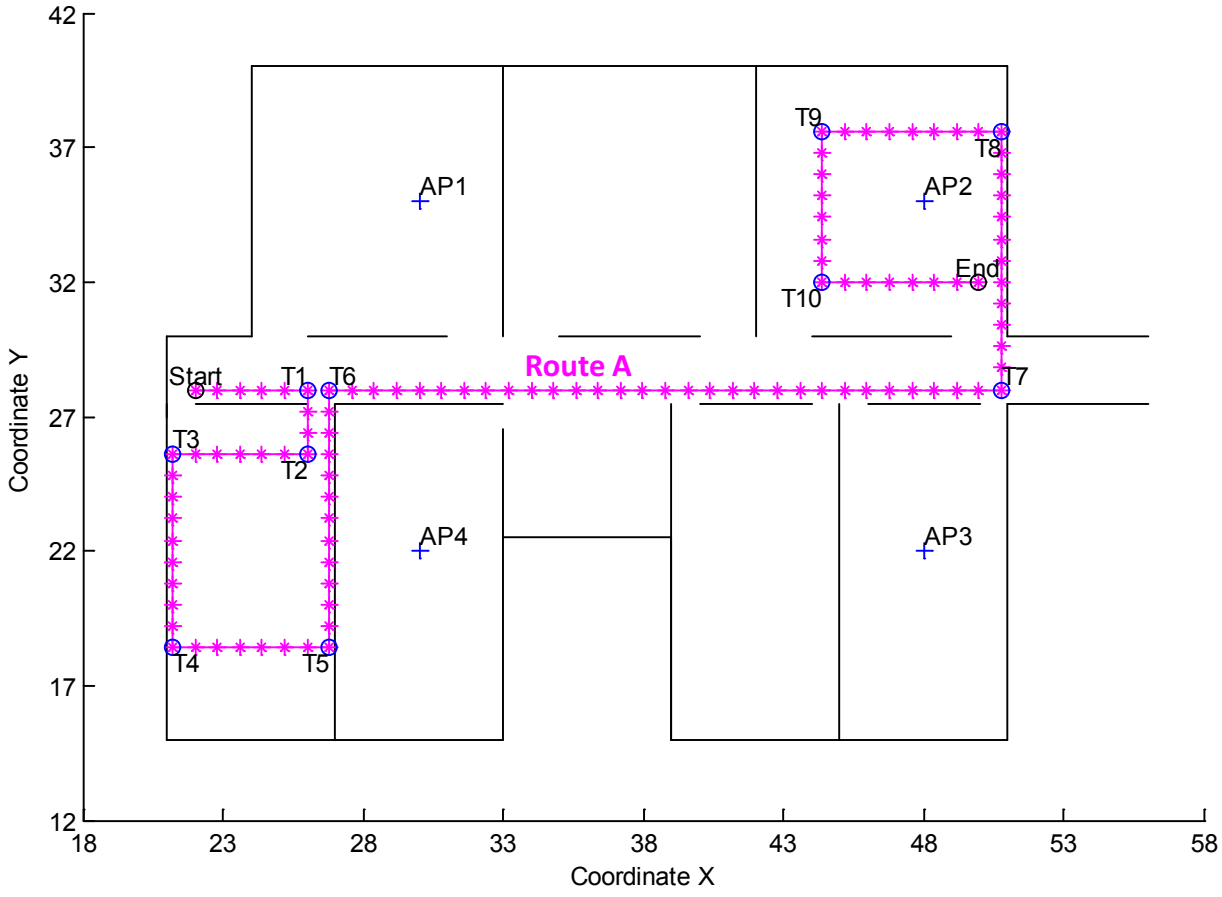


Fig. 4.14 Plane view of test-site no.1.

A total of ten trials were conducted. In each trial, the subject held a Nexus 7 at their waist and walked normally along the route, as illustrated in Figure 4.14 and Figure 4.15

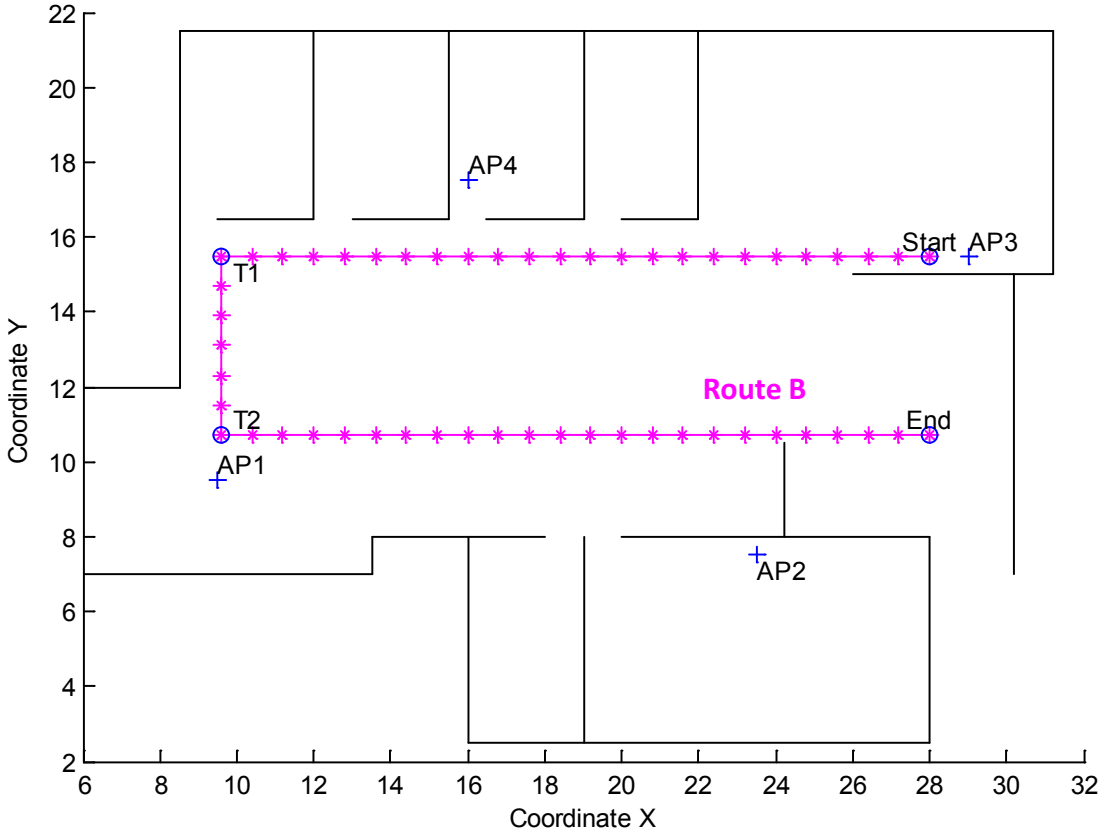


Fig. 4.15 Plane view of test-site no.2.

respectively, beginning from 'Start' until 'End'. The 'T1, T2, ... T10' indicate the positions where the pedestrian changed its heading. Note that some pauses and minor path deviations did occur because the pedestrians had to open doors as well as avoid bumping into other persons while walking along the same pathways. Nevertheless, subject was asked to follow the designated routes as closely as possible so that the occurrence of imprecise distance traveled can be minimized and impact of erroneous heading onto PDR's performance can be apparently observed. The primary purpose of having the routes designated so was to test the viability of the RSSI based heading correction and hence examine the improvement it could bring to the conventional PDR. Table 4.2 and Table 4.3 tabulate the corresponding route's straight paths and the actual number of steps performed in each path. Nonetheless, slightly miscounted number of steps is inevitable due to inaccurate step detection based on noisy inertial data. The ground-truth was determined by marking the time-stamp of every new position along the route.

The selection of positioning methods for a fair comparison in terms of positioning performance is rather difficult as the existing methods are uniquely designed to work optimally

Table 4.2 Route A in test-site no.1

Path No.	Segment of Route	No. of steps taken
1	Start – T1	5
2	T1 – T2	3
3	T2 — T3	6
4	T3 —- T4	9
5	T4 —- T5	7
6	T5 —- T6	12
7	T6 —- T7	30
8	T7 —- T8	12
9	T8 —- T9	8
10	T9 — T10	7
11	T10 – End	7

Table 4.3 Route B in test-site no.2

Path No.	Segment of Route	No. of steps taken
1	Start – T1	23
2	T1 – T2	6
3	T2 – End	23

with specific requirements. For instance, the Fingerprinting or landmarks based methods can be highly reliable only when the number of references is high which would require laborious and time-consuming site-surveys while some exploit external information like nearby pedestrians, indoor map and GPS. In short, better positioning performance comes with greater requirement in several aspects like available information, hardware and preparation work. The proposed method requires minimal information (i.e. RSSI, inertial and magnetic measurements only) from the pedestrian's device alone and no site survey. In this regard, the proposed method is similar to the ones proposed in [49] and [35]. For comparison purposes, a total of six positioning methods were considered in estimating the subject's positions throughout the routes, and they are listed as follows:

1. ***pPDR***: Sole PDR, as detailed in **Chapter 3.6**.
2. ***aTRI***: Tri-Lateration based on RSSI, as detailed in **Chapter 3.7.1**.
3. ***pSIR***: Fusion of PDR and RSSI via Sampling Importance Resampling Particle Filter, as detailed in **Chapter 3.7.2**.
4. ***pMCMC***: Fusion of PDR and RSSI via Markov Chain Monte Carlo sampling, as detailed in **Chapter 3.7.3**.
5. ***pKF***: Fusion of PDR and Tri-Lateration via Kalman Filter, as detailed in **Chapter 3.7.4**.
6. ***PM***: *pPDR* with proposed heading correction scheme.

Impact of varying number of deployed APs

As all the selected positioning methods excluding *pPDR* make use of the RSSI data, their respective robustness can thus be assessed based on their positioning performance under the impact of varying availability and density of RSSI data. The availability and density of RSSI data are proportional to the number of Wi-Fi APs deployed at the test-site. The number of deployed APs was varied from (originally) 4 units to 2 units according to the arrangements shown in Table 4.4. All the independent variables of respective methods (e.g. sample size, process noise covariance, and measurement noise covariance) were empirically configured to yield seemingly the best estimation results. In this work, they were configured only for the original cases whereby 4 units of APs were available, as depicted by Figure 4.14 and Figure 4.15. As for other cases whereby the number of APs was less than 4, the independent variables were not configured accordingly. This was intended to simulate the situations where the indoor positioning system, which is tuned beforehand under ideal condition, continues to work even under unanticipated circumstances like sudden breakdown of certain AP at the site.

Table 4.4 Arrangements of Wi-Fi APs considered during experiment

Arrangement No.	Sources of RSSI considered
1	AP1, AP2, AP3, AP4
2	AP1, AP2, AP3
3	AP1, AP3, AP4
4	AP1, AP2, AP4
5	AP2, AP3, AP4
6	AP1, AP2
7	AP1, AP3
8	AP1, AP4
9	AP2, AP3
10	AP2, AP4
11	AP3, AP4

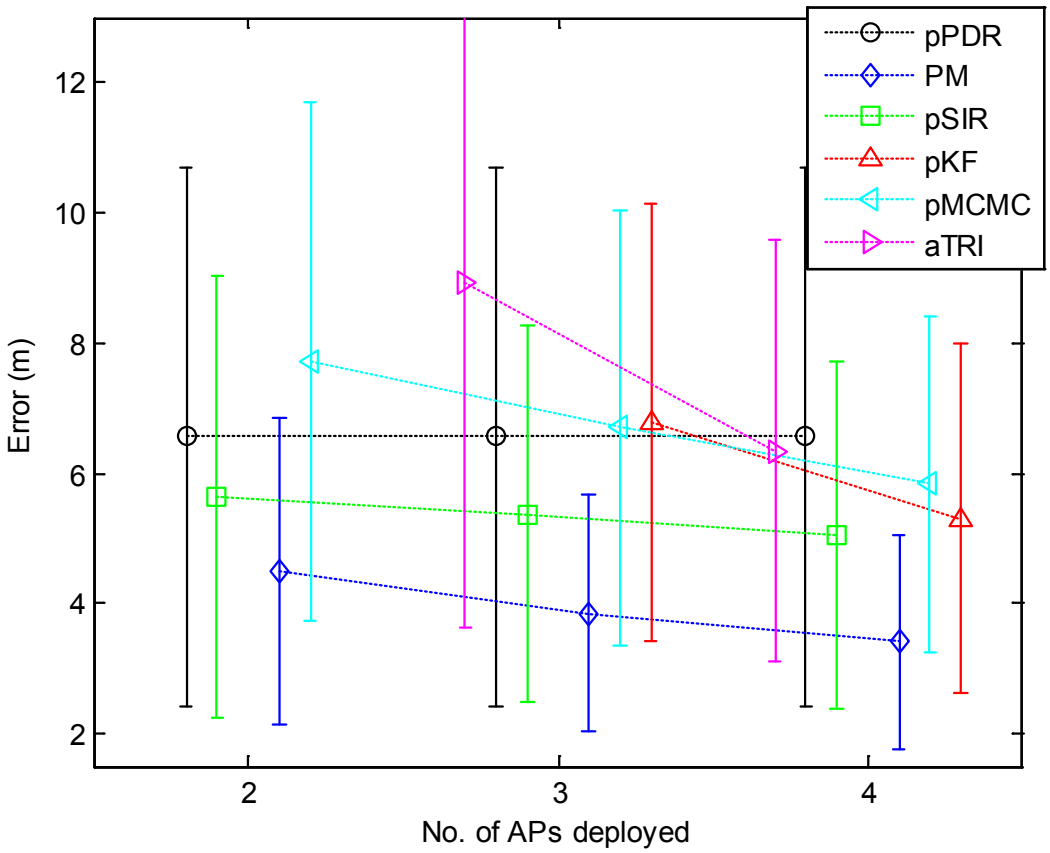


Fig. 4.16 Route A: Methods' errors versus No. of APs deployed.

From both Figure 4.16 and Figure 4.17, one can be observe that: (i) While *pPDR* remains unaffected as the number of APs decreases because it's independent of RSSI data, the accuracy of all the other methods deteriorates. From the steepness of change in average error when the number of APs decreases, it is apparent that the affected methods, arranged in the order from highest to lowest degree of severity, are *aTRI*, *pKF*, *pMCMC* and lastly *pSIR* and *PM*. It is unsurprising that *aTRI* suffered the most from reduced number of APs because it relies on solely RSSI data, while other methods use both PDR and RSSI data to complement the estimation of pedestrian's position. Recall that the outcomes of *aTRI* serve as measurements in *pKF*, therefore *pKF*'s deterioration rate is nearly as much as *aTRI*'s. Note that *aTRI* requires RSSI readings from at least three APs, therefore its performance can't be tested in cases where only two APs are deployed. As for *pMCMC* and *pSIR*, apart from their PDR component which is considerably reliable for short displacements, intakes of RSSI data are also controlled whereby only the RSSI that passes the threshold R_{th} is used in calculating the distance. This therefore somewhat tells why they are not as significantly influenced by RSSI as *aTRI* and *pKF* are. Even so, *pMCMC* falls behind *pSIR* probably due to its MCMC

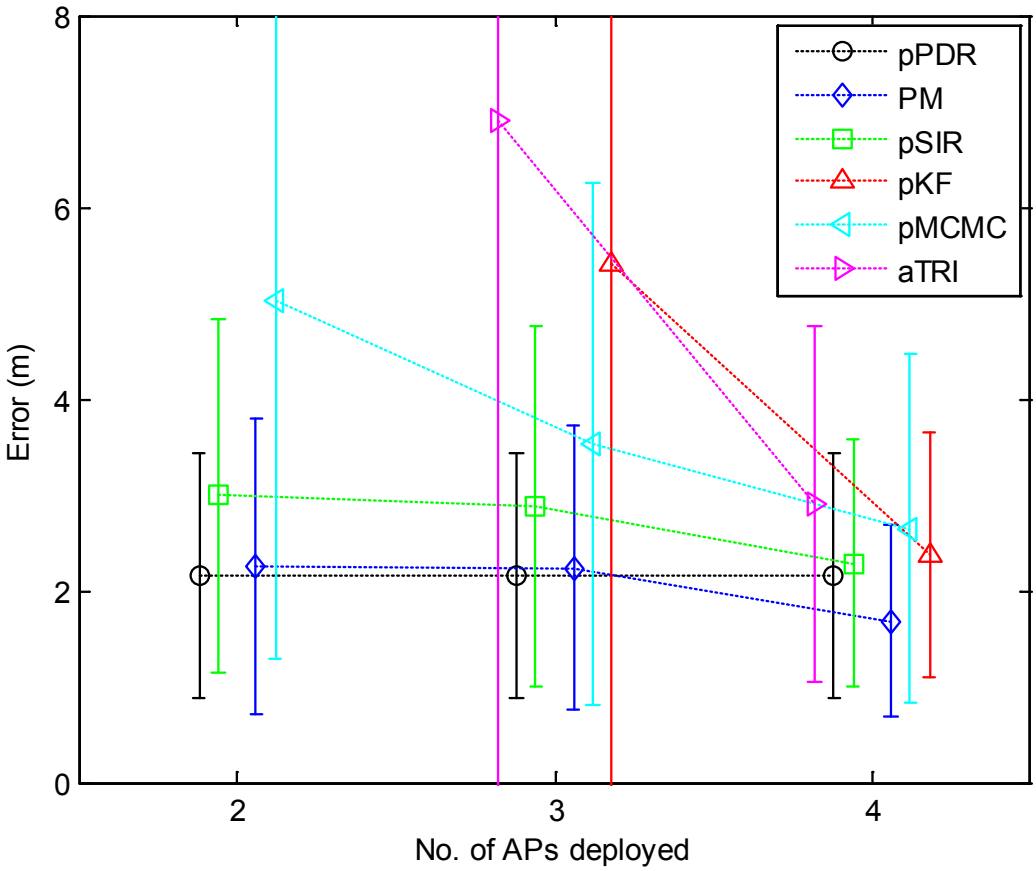


Fig. 4.17 Route B: Methods' errors versus No. of APs deployed.

sampling technique which, in comparison to SIR PF of *pSIR*, has higher demand for accurate RSSI data; otherwise poor or lack of RSSI data tends to cause indiscriminate acceptance of many subsequently generated samples that will eventually result in the average of the generated samples to deviate unreasonably far from previous position estimate.

(ii) The performance of *pPDR* is rather good in case of route B that it surpasses all methods except *PM*, as compared to its performance in the case of route A. One probable explanation is that route B is less complicated than route A by having lesser turns and shorter sum of distance traveled. Lesser turns made by pedestrian implies lower probability that estimated heading ϑ gets jeopardized by imprecisely detected change in pedestrian's heading $\Delta\theta$. Shorter distance traveled along a straight path implies lesser amount of accumulated error caused by inaccurate ϑ as the pedestrian advances further. In addition, though inaccurate $\Delta\theta$ is undesired, sometimes it does accidentally result in correct ϑ. Figure 4.18 illustrates an example of such phenomenon. Therefore, it is safe to conclude that PDR's reliability is unpredictable because the sensors' inherent biases might neutralize the previous flawed estimations (i.e. distance traveled and heading) by chance. Being the primary key to steadily

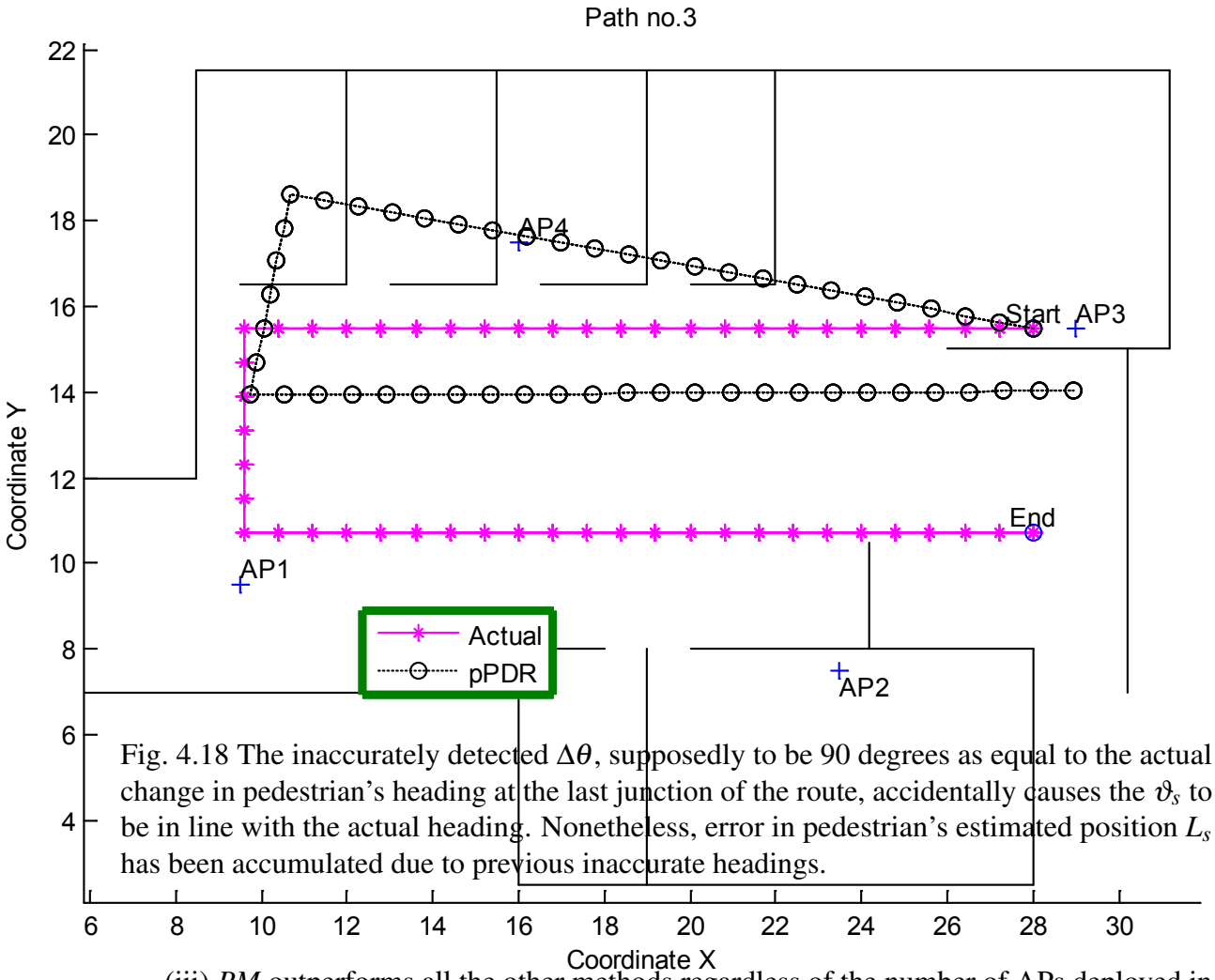


Fig. 4.18 The inaccurately detected $\Delta\theta$, supposedly to be 90 degrees as equal to the actual change in pedestrian's heading at the last junction of the route, accidentally causes the ϑ_s to be in line with the actual heading. Nonetheless, error in pedestrian's estimated position L_s has been accumulated due to previous inaccurate headings.

(iii) *PM* outperforms all the other methods regardless of the number of APs deployed in both cases, while being comparable with *pPDR* in case of route B. As previously mentioned, *PM* is actually *pPDR* with conditionally improved headings, thus it should portray an enhanced version of *pPDR*. The reason why *PM* is comparable with *pPDR* in the case of route B as shown in Figure 4.16 is that the PDR's accuracy had not deteriorated until the stage where the heading correction can yield significant improvement. On the contrary, *PM* excels in the case of route A (as shown in Figure 4.17) where *pPDR* has a 6.56 m mean error with 4.15 m standard deviation which is considerably worse than its 2.15 m mean error with 1.28 m standard deviation achieved in the case of route B. By triggering heading correction at single straight path namely path no.7 of route A, *PM* manages to improve PDR's mean error up to 49 %. *PM*'s performance can be more comprehensively demonstrated by its propagation of average error over the route A, as shown in Figure 4.19 where the error is the absolute difference between actual position and estimated position.

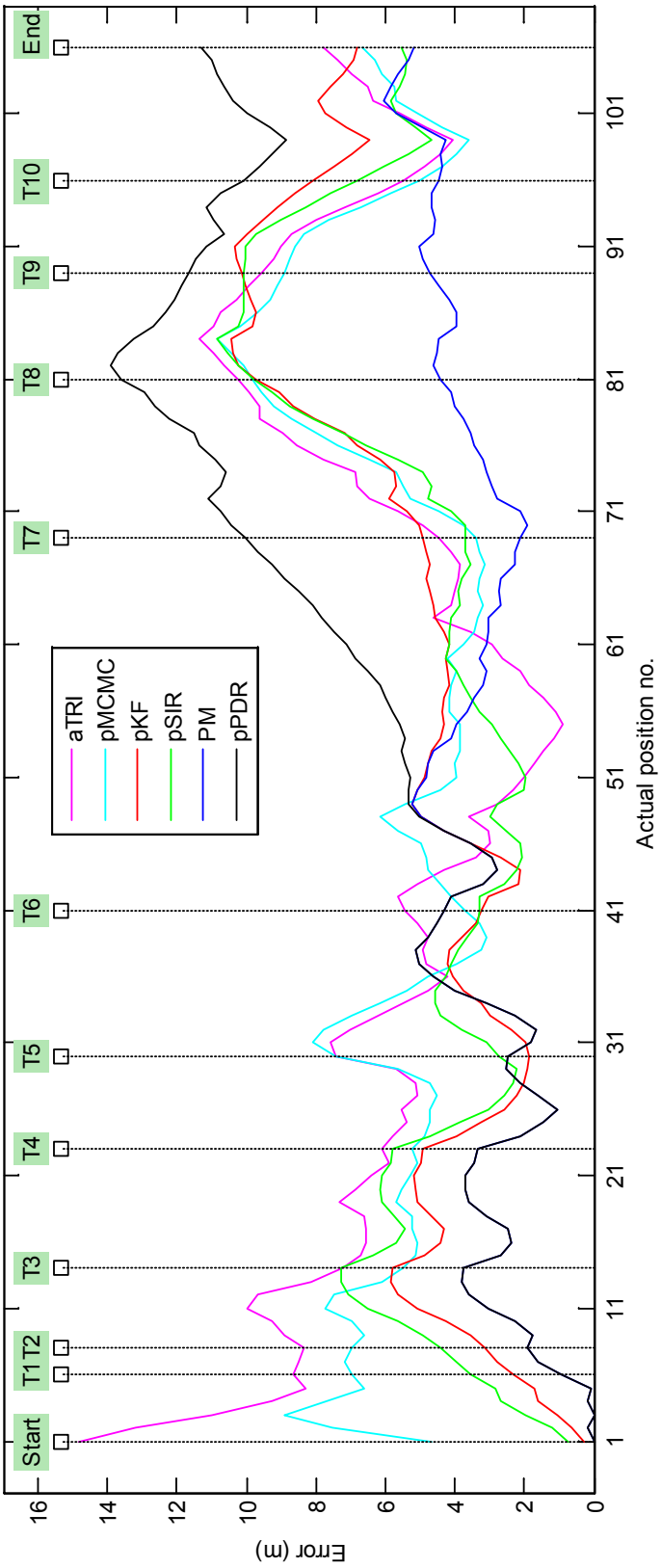


Fig. 4.19 Propagation of average error over the route A

From Figure 4.19, it can be observed that, beginning from the 'Start', *aTRI* being the only method which relies on solely RSSI data suffers immensely probably due to poor RSSI data while the deterioration of other methods is comparatively gradual because of their PDR component hence making them more resilient towards erratic RSSI data. As the very beginning heading is given, *PM* being the pure PDR and unaffected by RSSI at early stages like the *pPDR* has the slowest rate of deterioration as the pedestrian advances further along the route. During the period between 'T6' and 'T7' however, considering *aTRI*'s error has been significantly improved, it seems that the RSSI data have become somewhat useful to result in improvement on the accuracy of other RSSI-dependent methods while *pPDR*'s error escalates primarily due to erroneous headings. Nonetheless, after 'T7', the RSSI-dependent methods' accuracy fluctuates as the reliability of RSSI data does. Unlike *pPDR*, *PM* maintains its steady accuracy even after 'T6' because the interval between'T6' and 'T7' (i.e. path no.7) is where the new headings derived from orientation of regression line come into effect. As a result of that, the pedestrian's headings since then were also improved, as shown in Figure 4.20 which depicts the average discrepancies between actual headings and estimated headings at each path of route A. For example, as shown in Figure 4.21a which depicts one of the experiment results, *PM* as well as *pPDR* gradually deviates from the actual route due to mere PDR with miscounted steps and primarily erroneous heading; yet, after the heading ϑ at path no.7 gets aligned with the orientation of the regression line (which is depicted in Figure 4.21b), *PM* remains comparatively close to the actual route.

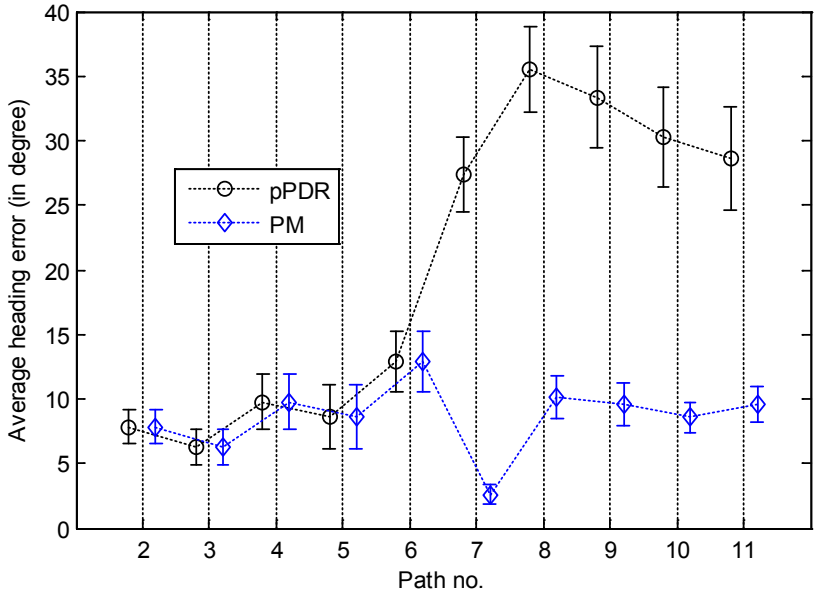
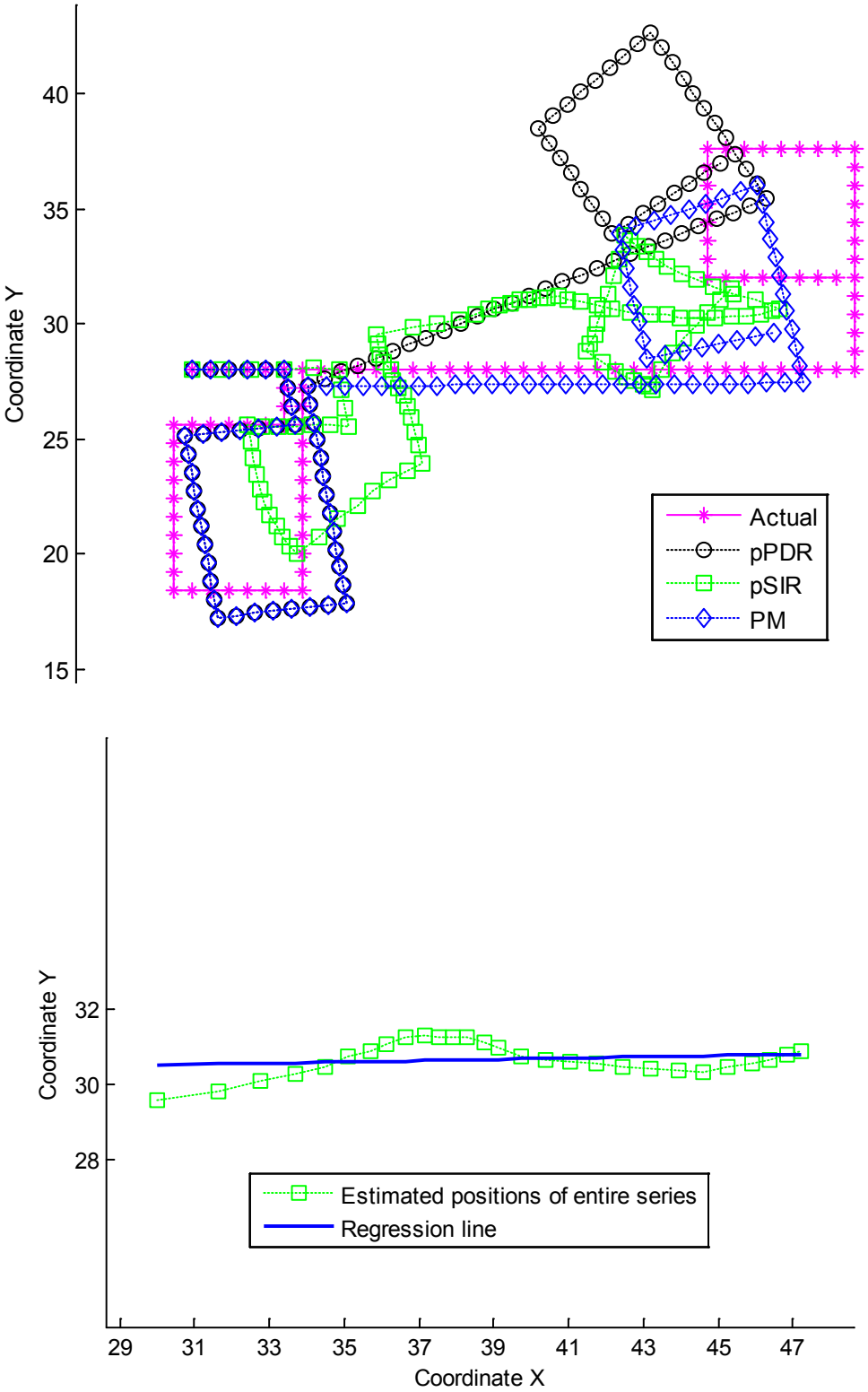


Fig. 4.20 Average heading error at each path of route A.



(b) Derived regression line for path no.7 of route A.

Fig. 4.21 An example of experiment results.

Heading Derivation based on Linear Regression

Table 4.5 and Table 4.6 tabulate the average heading errors for Route A and Route B respectively. The heading error is the scalar discrepancy between actual heading and heading derived based on linear regression (of *series*' positions estimated by respective methods). Note that the actual headings were the true azimuths for the designated routes. From Table 4.5 and Table 4.6, it is apparent that linear regression of *series*' positions estimated by all methods except *pPDR* result in average heading errors (highlighted in green) lower than those of *pPDR* (highlighted in grey) especially when the path's length is long. Recall that path no.7 is the longest straight path of route A, and path no.1 and no.3 are among the longest straight path of route B. As illustrated by a trial's result shown in Figure 4.22, the route estimated by *aTRI* itself seemingly deviates far from the actual route, yet the regression lines somewhat resemble the true headings of route B. Some additional results are depicted in Figure 4.23. Nonetheless, the accuracy of such heading derivation approach deteriorates as the number of the APs decreases. Even so, the statistics shown have demonstrated that, in occasion of long straight path, this technique may be feasible as an alternative to the conventional heading estimator of PDR. Note that this scheme is basically PDR with conditional heading correction; PDR is primarily used to estimate the positions of the pedestrian as he/she walks; correction of estimated heading occurs only when a long straight trajectory is identified; Figures 4.18, 4.21, 4.22 and 4.23 apparently show that heading correction significantly enhances the PDR-estimated positions.

Table 4.5 Average scalar discrepancies between actual headings and headings derived based on linear regression of *series*' positions estimated by respective methods in the case of route A. Note that the values highlighted in grey are the heading errors of *PM* when the heading correction is not applied; the values highlighted in green are smaller than those highlighted in grey.

Path No.	Average Heading Error, (degree)				
	pPDR	aTRI	pKF	pMCMC	pSIR
2	7.9	40.0	18.1	38.6	29.1
3	6.3	8.1	6.9	21.4	7.4
4	9.8	52.1	12.1	32.7	8.3
5	8.7	94.9	32.0	84.7	46.7
6	12.9	75.0	23.6	50.8	24.5
7	27.4	2.8	6.8	7.3	3.1
8	35.6	46.3	55.4	54.9	58.6
9	33.4	92.5	52.6	91.5	77.4
10	30.3	141.0	39.9	126.9	65.5
11	28.6	28.5	32.9	12.7	44.4

(a) When No. of APs is 4

Path No.	Average Heading Error, (degree)				
	pPDR	aTRI	pKF	pMCMC	pSIR
2	7.9	58.2	16.4	44.4	25.8
3	6.3	31.5	16.2	20.5	7.2
4	9.8	73.1	16.4	49.2	10.1
5	8.7	98.3	44.7	82.3	27.0
6	12.9	72.4	24.1	59.5	18.2
7	27.4	25.6	7.0	14.7	7.9
8	35.6	66.6	58.5	53.1	63.1
9	33.4	92.4	54.0	84.0	74.7
10	30.3	117.4	39.4	124.9	59.2
11	28.6	31.3	31.8	17.8	45.3

(b) When No. of APs is 3

Table 4.6 Average scalar discrepancies between actual headings and headings derived based on linear regression of *series*' positions estimated by respective methods in the case of route B. Note that the values highlighted in grey are the heading errors of *PM* when the heading correction is not applied; the values highlighted in green are smaller than those highlighted in grey.

Path No.	Average Heading Error, (degree)				
	pPDR	aTRI	pKF	pMCMC	pSIR
1	7.2	7.0	5.9	6.3	5.7
2	10.2	58.5	44.7	26.3	23.8
3	12.3	11.6	10.3	7.4	5.6

(a) When No. of APs is 4

Path No.	Average Heading Error, (degree)				
	pPDR	aTRI	pKF	pMCMC	pSIR
1	7.2	22.0	16.0	10.0	7.5
2	10.2	62.3	53.6	40.2	36.8

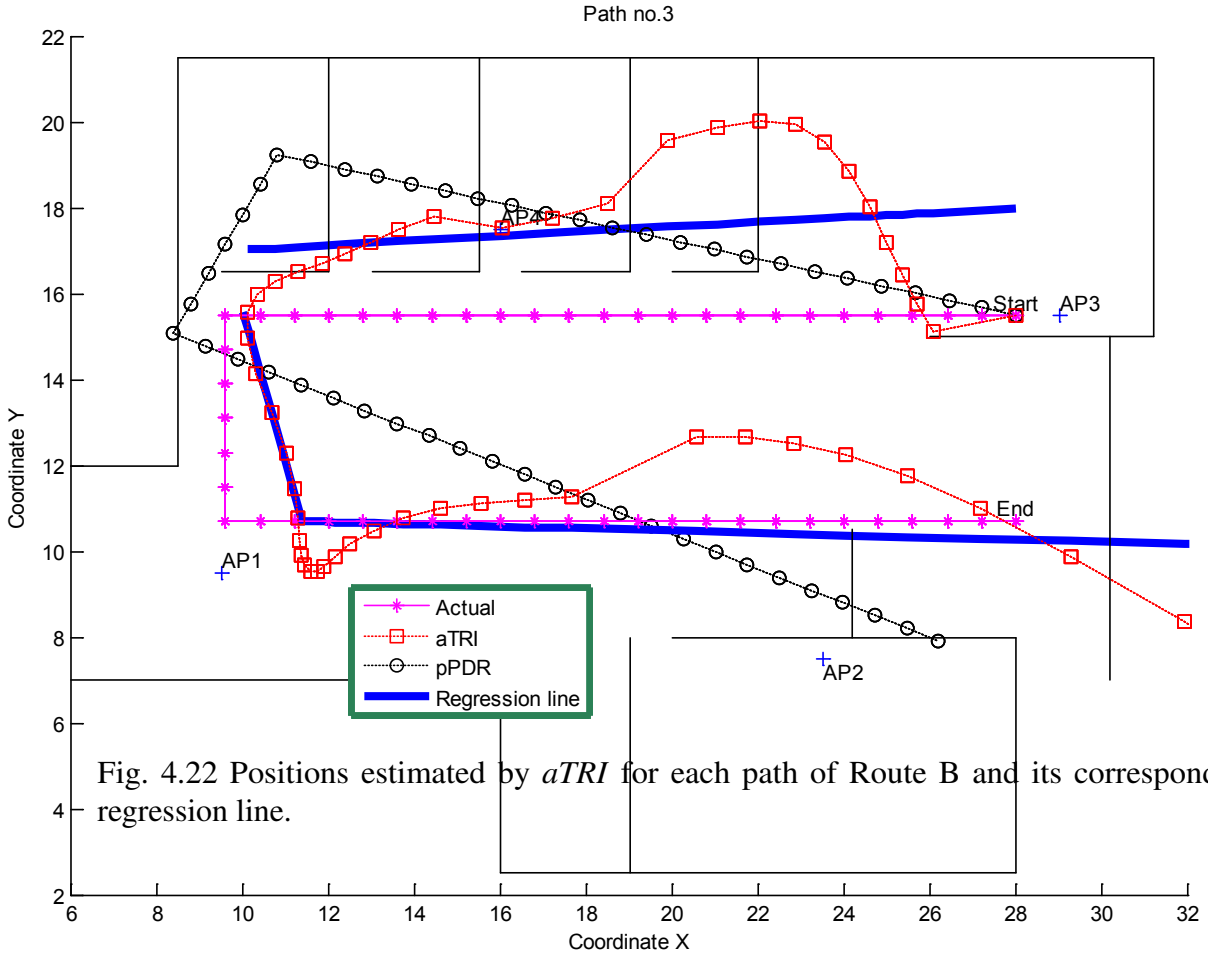
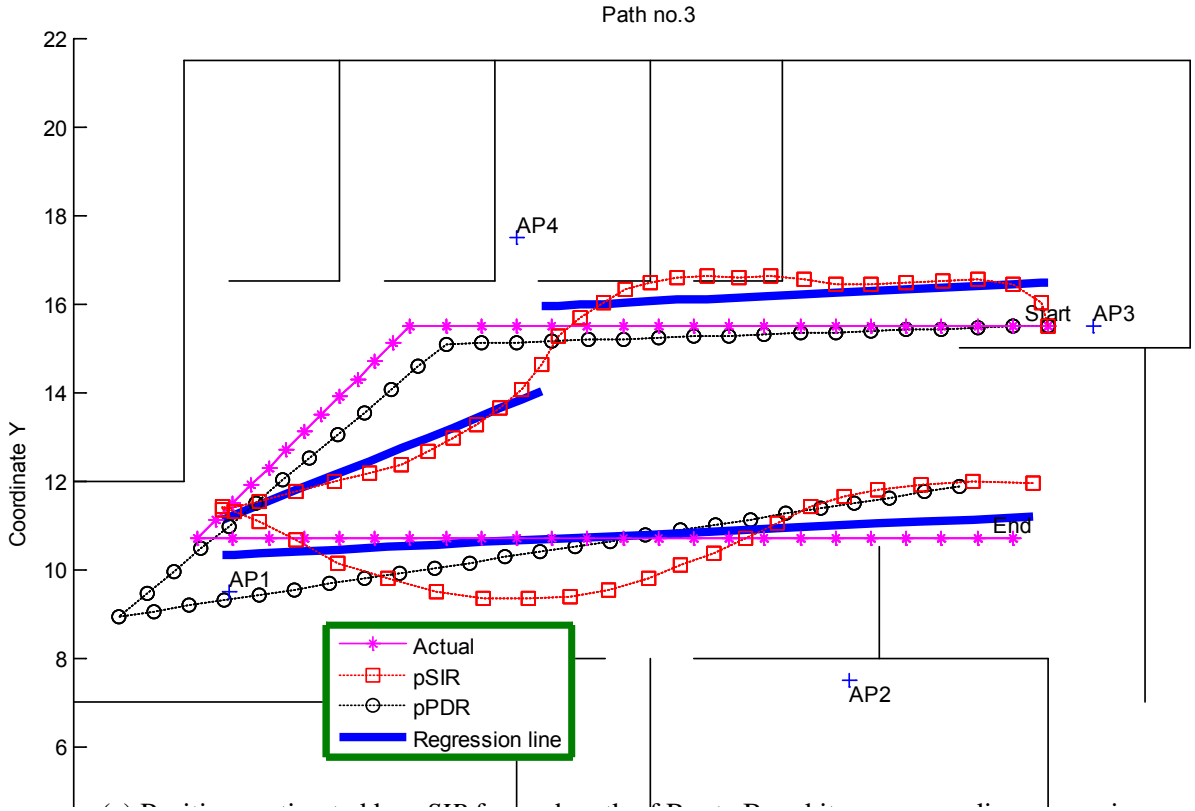
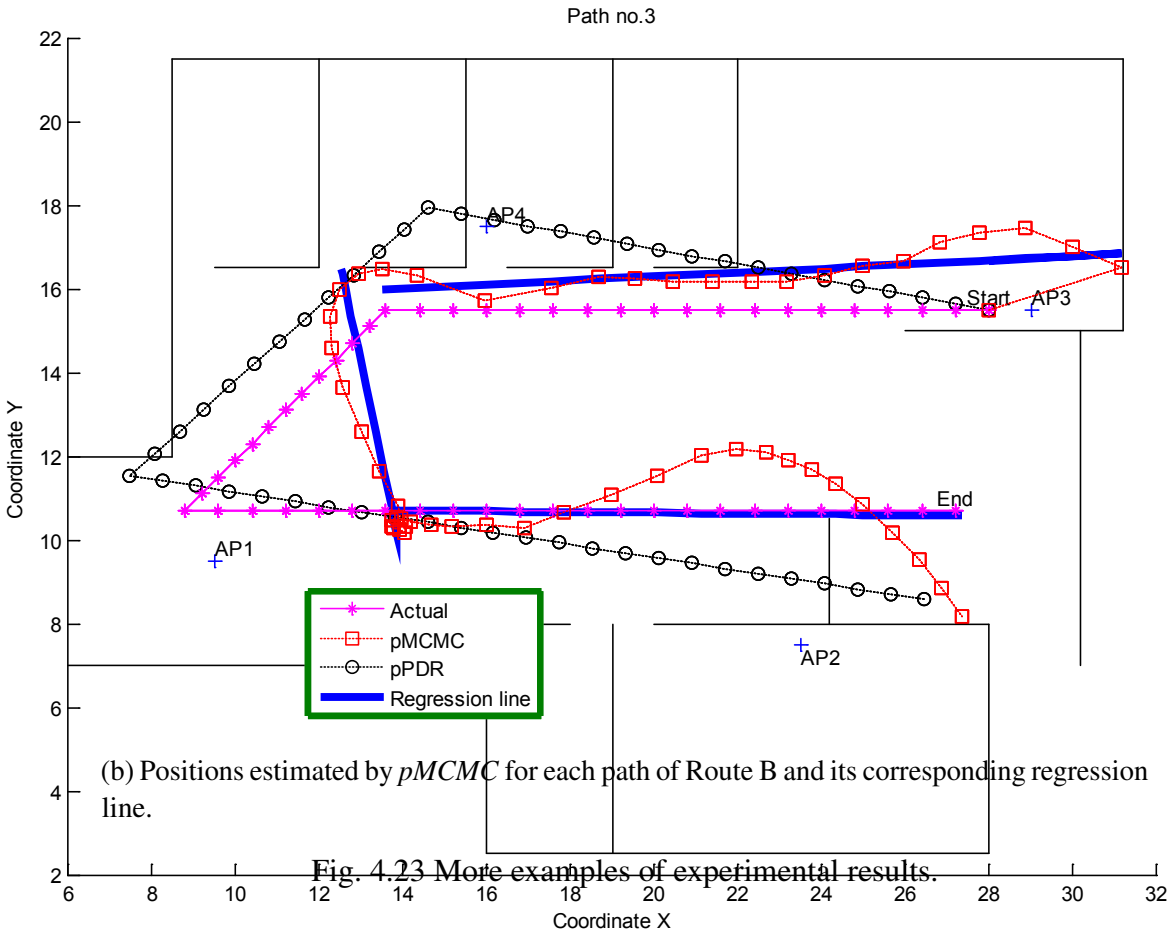


Fig. 4.22 Positions estimated by *aTRI* for each path of Route B and its corresponding regression line.





(b) Positions estimated by *pMCMC* for each path of Route B and its corresponding regression line.

Fig. 4.23 More examples of experimental results.

4.3 Summary

This chapter presents two schemes that enhance the positioning accuracy of smart-device -based PDR by utilizing the RSSI of Wi-Fi signals. Based on merely the sensory data from the device's embedded sensors, conventional PDR is employed for its simplicity as well as its high but short-term reliability. To ensure PDR's high reliability for long-term positioning, its errors must be rectified whenever appropriate.

The first scheme aims to refine the PDR-estimated position of pedestrian, by ensuring that the estimated position falls within certain region of corresponding Wi-Fi AP according to the obtained RSSI value as well as inputted compass bearing. Admittedly, this scheme may not be viable in situations where human intervention is not allowed. The second scheme aims to correct the PDR-estimated headings based on linear regression, and subsequently update the corresponding PDR-estimated positions. The heading correction, however, gets triggered only upon detecting a long straight PDR-estimated trajectory which can be commonly observed in pedestrian's daily movements, e.g. moving along long straight pathways in university, mega-mall, etc.

Both schemes had been experimentally tested. The test results have demonstrated: firstly, the feasibility of the proposed schemes; secondly, PDR with proposed schemes outperform some existing PDR-based positioning methods in terms of positioning accuracy as well as robustness towards lack of Wi-Fi APs.

Chapter 5

Collaborative Indoor Positioning based on Directed Graph

As mentioned in **Chapter 2.4**, many indoor positioning methods have already been proposed leveraging the collaboration among pedestrians to achieve more refined positioning. In contrast to those existing collaborative methods, the one proposed in this work first estimates the pedestrians' positions by using Pedestrian Dead-Reckoning or Tri-Lateration, and then refines them via an iterative correction process with measurements derived from the Directed Graph. The Directed Graph is constructed to express the spatial relations among the pedestrians and site's Wi-Fi APs. The procedure of the proposed collaborative positioning method is briefly depicted in Figure 5.1.

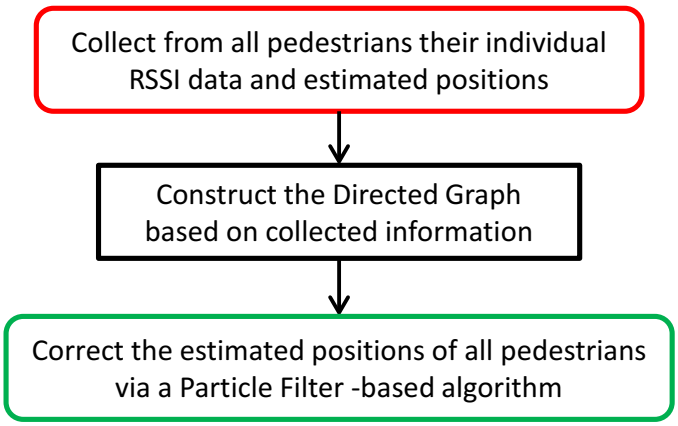


Fig. 5.1 Procedure of proposed collaborative positioning method.

5.1 Construction of Directed Graph

An indoor crowded scenario where there are N_p pedestrians is considered. There are also N_a units of Wi-Fi APs sparsely deployed at the site. The pedestrians can detect the surrounding pedestrians and APs by observing the RSSI data via Bluetooth and Wi-Fi capabilities of smart-devices. The true positions of APs are assumed known while pedestrians' may vary over time. Note that the pedestrian and the AP are now termed Mobile Node (MN) and Beacon Node (BN), respectively.

Let a denote the index of MN and b denote the index of the RSSI source which can be either MN or BN. b-th node is considered a *neighbour* to a-th node when the observed RSSI, $R_{a,b} \geq R_{th}$ where R_{th} is an RSSI threshold which is to be defined empirically so that a-th node is considered near to b-th node. The distance between the nodes, D can be derived from the observed RSSI via Equation (4.2). Though the estimate is mostly inaccurate in practice, it is somewhat representative for short distances when RSSI values are strong [31, 74].

The relations among MNs and BNs at time t is expressed by a weighted directed graph $G = (V, E)$ where V is the set of vertexes and E is the set of edges. The vertexes are comprised of N_p MNs and N_a BNs, and the edges are the ordered pairs of vertexes. The weight of each edge is their corresponding D. Then, the said graph can be represented by a weighted adjacency matrix,

$$M_t = \begin{pmatrix} S_{1,1} & \dots & S_{1,N_v} \\ \vdots & \ddots & \vdots \\ S_{N_v,1} & \dots & S_{N_v,N_v} \end{pmatrix}$$

where N_v is the summation of N_a and N_p; $S_{i,j}$ denotes the shortest distance from i-th vertex to j-th vertex. When i equates j, $S_{i,j}$ is zero. $S_{i,j}$ is infinite when there is no path between i-th vertex and j-th vertex. Figure 5.2 depicts an example of directed graph for four nodes and its corresponding weighted adjacency matrix. In Figure 5.2, the number of red arrows going outward from each node is equivalent to the number of *neighbours* each node has; though node '1' is distant and undetectable to node '4', the distance between them can be deemed as $S_{4,1} = D_{4,3} + D_{3,2} + D_{2,1}$. The derived $S_{i,j}$ is not necessarily the actual displacement between the nodes especially when there is no direct path between them. Nevertheless, it can be used to somewhat bound the absolute difference between the nodes' estimated positions. Therefore, in such way, even those distant pedestrians besides the neighbouring ones could also be useful references in refining the estimated positions.

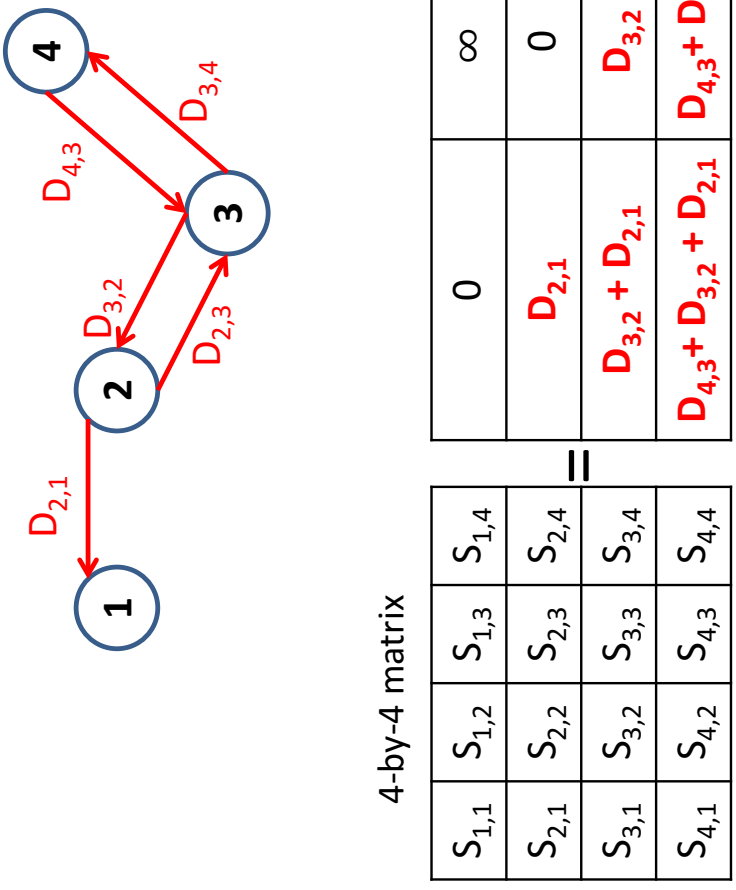


Fig. 5.2 An example of directed graph for four nodes and its corresponding weighted adjacency matrix, M

5.2 Correction Algorithm

Once the weighted adjacency matrix (M_t) at certain time t is ready, the correction of estimated positions of all N_p MN shall be executed. The correction of estimated positions is usually an iterative process, as depicted in Figure 5.3 where $P_{i,x}$ denotes the estimated position of i-th MN at x-th iteration; and γ is an empirically defined parameter. Note that $P_{i,0}$ for $i = 1,2,3,\ldots,N_p$ are estimated by employing Tri-Lateration or PDR; the algorithms of both Tri-Lateration and PDR are explained in **Chapter 3**.

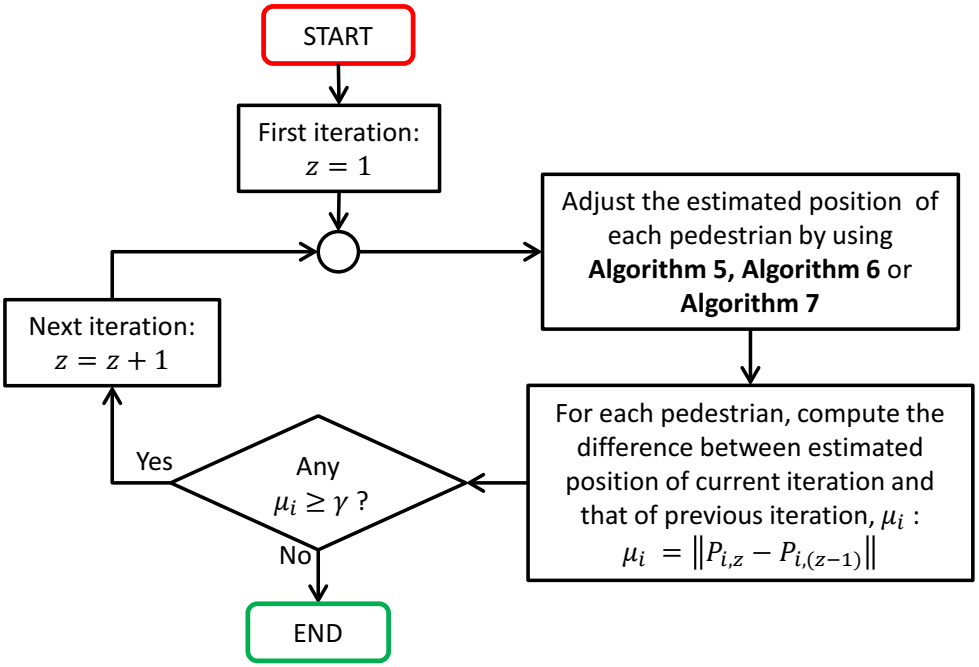


Fig. 5.3 Flowchart of Correction Process.

At every new iteration during the correction process, the estimated positions of all N_p MN can be adjusted by using any of the three algorithms depicted by **Algorithm 5**, **Algorithm 6** and **Algorithm 7** respectively. All these three algorithms have two things in common. Firstly, all finite and non-zero $S_{i,j}$ obtained from M_t are utilized to somewhat bound the absolute difference between P_i and corresponding j-th node. This means that besides neighbouring nodes, even the distant ones could also be references. Secondly, BN is more influential than MN in finalizing P_i. The reason behind is that the BN's position, unlike the MN's, remains unaffected throughout the iterations and therefore serves to ground the correction process which may otherwise converge with a significant bias. In fact, **Algorithm 7** is the most preferred choice in refining the estimated positions, because **Algorithm 5** and **Algorithm 6** are basically its predecessor versions which yield comparatively poorer positioning performance.

Algorithm 5 Adjust the estimated position of i-th MN

I is an empty set
for $j \leftarrow 1$, number of columns of M_t **do**
  if $S_{i,j} \neq 0$ and $S_{i,j} \neq \infty$ **then**
    if $\|P_{i,(z-1)} - P_{j,(z-1)}\| > S_{i,j}$ **then**
      $\theta \leftarrow$ the positive angle between vector $\overrightarrow{P_{j,(z-1)}P_{i,(z-1)}}$ and positive x-axis
      $p \leftarrow P_{j,(z-1)} + S_{i,j} \times \begin{bmatrix} \cos\theta & \sin\theta \end{bmatrix}^{\mathsf{T}}$
      add p to I
    end if
  end if
end for
if I is empty **then**
  $P_{i,z} \leftarrow P_{i,(z-1)}$
else
  $P_{i,z} \leftarrow$ mean of I
end if
if any BN is a *neighbour* of i-th MN **then**
  $F \leftarrow$ position of corresponding BN
  $d \leftarrow$ RSSI-derived distance between i-th MN and corresponding BN
  $\theta \leftarrow$ the positive angle between vector $\overrightarrow{FP_{i,z}}$ and positive x-axis
  if $\|P_{i,z} - F\| > d$ **then**
    $P_{i,z} \leftarrow F + d \times \begin{bmatrix} \cos\theta & \sin\theta \end{bmatrix}^{\mathsf{T}}$
  end if
end if

The procedure of **Algorithm 5** can be summarized into three steps. Firstly, the distance between estimated position of i-th MN and estimated position of j-th node should not be greater than their corresponding shortest distance (i.e. $S_{i,j}$); else, the estimated position of i-th MN should be shifted radially towards the corresponding j-th node's estimated position. Secondly, if the estimated position of i-th MN needs to be shifted towards multiple j-th nodes, then the estimated position of i-th MN shall be the average of those shifted estimated positions. Lastly, if i-th MN has a neighbour which is an AP, then the distance between estimated position of i-th MN and corresponding AP's actual position should not be greater than the corresponding RSSI-derived distance; else, the estimated position of i-th MN shall be shifted radially towards the corresponding AP's actual position.

Algorithm 6 Adjust the estimated position of i-th MN

I is an empty set
for $j \leftarrow 1$, number of columns of M_t **do**
    if $S_{i,j} \neq 0$ and $S_{i,j} \neq \infty$ **then**
        $\theta \leftarrow$ the positive angle between vector $\overrightarrow{P_{j,(z-1)}P_{i,(z-1)}}$ and positive x-axis
        $p \leftarrow \left(P_{j,(z-1)} + S_{i,j} \times \begin{bmatrix}\cos\theta & \sin\theta\end{bmatrix}^\mathsf{T}\right)$
        add p to I
    end if
end for
$P_{i,z} \leftarrow$ mean of I
if any BN is a *neighbour* of i-th MN **then**
    $F \leftarrow$ position of corresponding BN
    $d \leftarrow$ RSSI-derived distance between i-th MN and corresponding BN
    $\theta \leftarrow$ the positive angle between vector $\overrightarrow{FP_{i,z}}$ and positive x-axis
    if $\|P_{i,z} - F\| > d$ **then**
        $P_{i,z} \leftarrow F + d \times \begin{bmatrix}\cos\theta & \sin\theta\end{bmatrix}^\mathsf{T}$
    end if
end if

The procedure of **Algorithm 6** can be summarized into three steps. Firstly, every non-zero and finite $S_{i,j}$ along the i-th row of M_t shall contribute a preliminary estimate which signifies the likely position of i-th MN. Secondly, the estimated position of i-th MN shall be the average of all the preliminary estimates. Lastly, if i-th MN has a neighbour which is an AP, then the distance between estimated position of i-th MN and corresponding AP's actual position should not be greater than their corresponding RSSI-derived distance; else, the estimated position of i-th MN shall be shifted radially towards the corresponding AP's actual position.

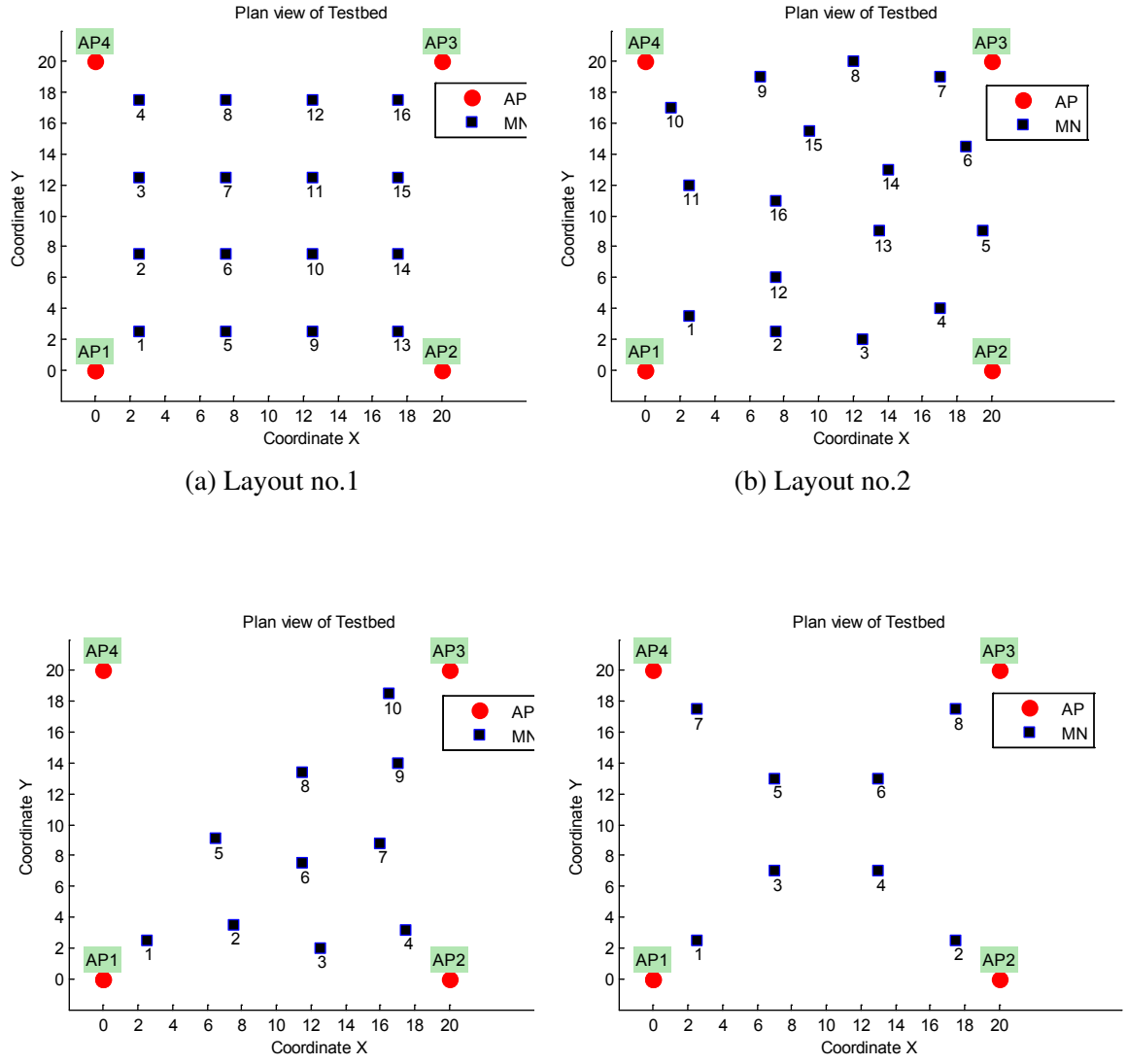
Algorithm 7 Adjust the estimated position of i-th MN

1: both A and B are empty sets
2: **for** $j \leftarrow 1$, number of columns of M_t **do**
3: **if** $S_{i,j} \neq 0$ and $S_{i,j} \neq \infty$ **then**
4: $\theta \leftarrow$ the positive angle between vector $\overrightarrow{P_{j,(z-1)}P_{i,(z-1)}}$ and positive x-axis
5: $I \leftarrow P_{j,(z-1)} + S_{i,j} \times \begin{bmatrix} \cos\theta & \sin\theta \end{bmatrix}^{\mathsf{T}}$
6: **for** $k \leftarrow 1$, desired number of samples **do**
7: $Q^{[k]} \leftarrow$ sample from a Normal distribution, $\mathcal{N}(I, \sigma)$
8: $w^{[k]} \leftarrow \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2(\sigma)^2}(S_{i,j} - \|Q^{[k]} - P_{j,(z-1)}\|)^2\right)$
9: **end for**
10: $w \leftarrow$ divide w by the sum of w
11: **for** $k \leftarrow 1, N$ **do**
12: $R^{[k]} \leftarrow$ randomly draw a sample from the weighted set of samples, i.e. $\{Q, w\}$, with probability proportional to its weight
13: **end for**
14: $I \leftarrow$ mean of R
15: **if** (j-th node is an BN) and (j-th node is a *neighbour* of i-th MN) **then**
16: add I to A
17: **else**
18: add I to B
19: **end if**
20: **end if**
21: **end for**
22: **if** B is empty **then**
23: $P_{i,z} \leftarrow$ mean of A
24: **else**
25: $P_{i,z} \leftarrow \alpha \times \big(\text{mean of } A\big) + \big(1-\alpha\big) \times \big(\text{mean of } B\big)$
26: **end if**

The procedure of **Algorithm 7** can be summarized into three steps. Firstly, every non-zero and finite $S_{i,j}$ along the i-th row of M_t shall contribute a preliminary estimate which signifies the likely position of i-th MN. Secondly, every preliminary estimate is processed via a custom Particle Filter (as detailed by lines no.6 to 14). Lastly, the estimated position of i-th MN is finalized by computing the weighted mean of all the preliminary estimates. Among all the preliminary estimates, only those which are acquired based on neighbouring AP contribute comparatively more "weights". The α shown in line no.25 is an empirically define parameter whose value ranges from 0 to 1. As BN is more influential than MN in finalizing P_i, therefore α is supposed to be greater than 0.5.

5.3 Evaluation I

The proposed collaborative method had been tested with four different layouts of testbed, as illustrated in Figure 5.4 where 'AP' denotes the actual locations of Wi-Fi APs and 'Actual' denotes the actual positions of pedestrians. Note that the Wi-Fi APs were considered BNs while pedestrians were considered MNs.



(a) Layout no.1

(b) Layout no.2

(c) Layout no.3

(d) Layout no.4

Fig. 5.4 Plane view of four different layouts of testbed

Each MN obtained the RSSI values from Wi-Fi signals emitted by BNs, and Bluetooth signals emitted by other MNs. From the obtained RSSI value (R), the corresponding distance (D) can thus be derived via a path loss model expressed by Equation (2.6) whose parameters'

values are shown in Table 3.3. The RSSI source whose RSSI value passed the R_{th} was considered as the *neighbour* of corresponding MN. Under this testbed, a node is considered *neighbour* of a MN only if the distance between them is not greater than 5 meters. Therefore, -60 dBM and -70 dBM were selected as R_{th} for Wi-Fi signal and Bluetooth signal respectively so that the distance estimated via the path-loss model was unlikely to be greater than 5 meters. For each layout of testbed, ten trials were conducted. For each trial, only the RSSI data at single time instance were used to compute the positions of all MNs. Note that all MNs' positions were estimated at the same time, by using the RSSI values which were collected at that instant. For comparison purposes, five different positioning methods were considered:

1. *TriA* : Each MN's position is estimated via Tri-Lateration by using the algorithm described in Chapter 3.4.1.

2. *TriB* : The estimated position of each MN is the average of their respective *TriA*-estimate and their *neighbours*' *TriA*-estimates.

3. *PmA* : *TriA*-estimates are processed by the proposed collaborative method which employs **Algorithm 5**.

4. *PmB* : *TriA*-estimates are processed by the proposed collaborative method which employs **Algorithm 6**.

5. *PmC* : *TriA*-estimates are processed by the proposed collaborative method which employs **Algorithm 7**.

The errors of all ten trials for all five different methods in respective layouts of tesbed are shown in Figure 5.5. The error metric is the absolute difference between the MN's estimated position and actual position. In Figure 5.5, each box-plot shows the distribution of $N_T \times N_m$ errors where N_T is the number of trials conducted and N_m is the number of MNs involved. It is worth mentioning that the mean of errors does not represent well the performance of a positioning method. Upper quartile, median and inter-quartile range should be the figures of merit instead. The lower the upper quartile or median, the higher the accuracy. The shorter the inter-quartile range, the higher the consistency. Therefore, from Figure 5.5, it is apparent that regardless of which layout, the five methods arranged in the order from lowest to highest positioning performance are *TriA*, *TriB*, *PmA*, *PmB* and *PmC*. Though insignificant, *PmA* does outperform both *PmB* and *PmC* in layout no.4. More comparisons among *PmA*, *PmB* and *PmC* shall be discussed later.

Number of references vs. Number of *neighbours* vs. Errors

While correcting the estimated position of an MN, the proposed method basically considers any other nodes – which can be directly or indirectly linked to in the graph, as references. Note that the nodes which are directly linked to MN are *neighbours*, while the indirectly linked ones are non-*neighbours*. The number of *neighbours* each MN has might vary at any time t because it depends on whether the RSSI values given by surrounding nodes pass the RSSI thresholds. The number of references each i-th MN has is equivalent to the number of j-th nodes which contribute non-zero and finite $S_{i,j}$. Figure 5.6, Figure 5.7, Figure 5.8 and Figure 5.9 summarize some information regarding the ten trials conducted in respective layouts. Each of these four Figures consists of three charts: firstly, the top chart shows the number of references made available from the graph for every trial, where the red horizontal line denotes the supposed number of available references; secondly, the middle chart illustrate the distribution of number of *neighbours* for every trial; thirdly, the bottom chart presents the distribution of errors yielded from three different proposed methods in every trial. As for the top charts, the fact that the number of references made available by the constructed graph fail to meet the supposed value in some trials implies that the Bluetooth RSSI data were not reliable enough to ensure that all MNs can detect their respective *neighbours* successfully. Nevertheless, by comparing the top charts with the middle charts, we can observe that the number of available references in most of the trials approach/reach the supposed value even though the number of *neighbours* a MN has may be as low as 1. Note that the number of available references in trial no.8 for layout no.3 exceeds the maximum value by 1 because 'AP4', owing to erratic Wi-Fi RSSI, was accidentally considered as *neighbour* of 8-th MN. By comparing the middle charts with the bottom charts, it seems that there is no correlation

racy

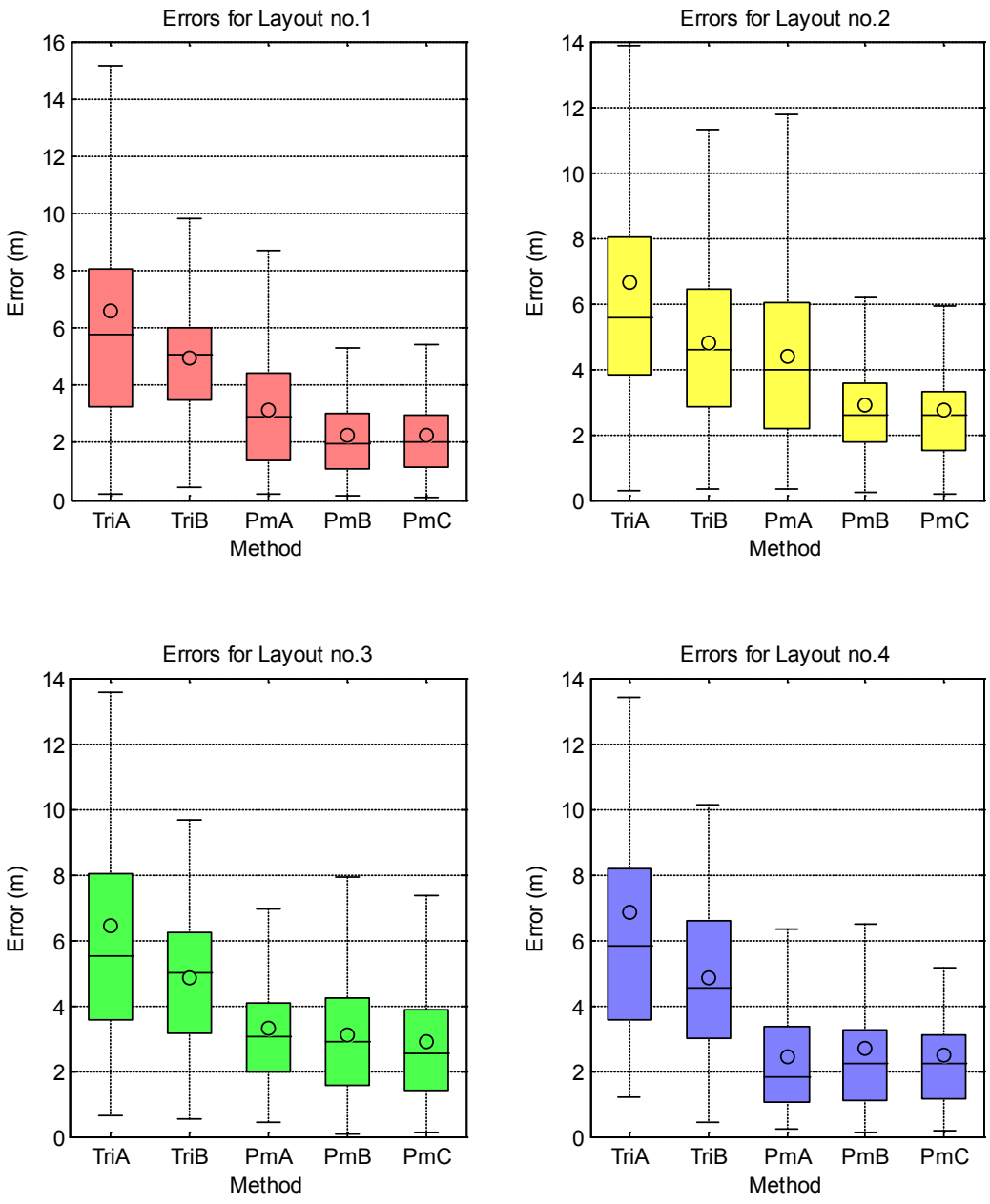


Fig. 5.5 Errors of five different methods for four different layouts of testbed.

found between the number of *neighbours* and the error of proposed methods. The consistency of numbers of *neighbours* does not necessarily imply the consistency of errors. For instance, as referring to Figure 5.9, the errors of proposed methods in trial no.3 are comparable to those in trial no.8 although the former trial has comparatively rather inconsistent number of *neighbours*. The larger the interquartile range, the greater the inconsistency of values. Besides, the error is not proportional to the number of *neighbours*. For example, as referring to Figure 5.9, the difference between the errors in trial no.8 and those in trial no.9 is apparent despite that they have almost identical number of *neighbours*. Such phenomenon is expected because the majority of the *neighbours* each MN has are fellow MNs whose estimated positions are subject to changes during the correction process.

Errors of respective MNs

In Figure 5.10, Figure 5.11, Figure 5.12 and Figure 5.13, the top chart shows the numbers of *neighbours* every MN had throughout all the trials in respective layouts, while the bottom chart depicts the corresponding errors for every MN throughout all the trials in respective layouts. By carefully comparing the top charts with the bottom charts, one can observe that the MNs which are near to any AP (i.e. MNs no.1,4,13&16 in layout no.1, MNs no.1,4,7&10 in layout no.2, MNs no.1,4&10 in layout no.3, and MNs no.1,2,7&8 in layout no.4.) tend to have lesser *neighbours* as well as smaller errors than the remaining MNs. This further implies that greater number of *neighbours* does not necessarily result in lower errors. Other than being affected by erratic RSSI, the reason why these MNs tend to have relatively lesser *neighbours* is that they, as can be seen from Figure 5.4, are situated at the border and not really surrounded by other MNs. These MNs have relatively lower errors probably because they have an AP as their *neighbour*; recall that the proposed methods put more weights on neighbouring AP than on neighbouring MN in correcting the estimated position of MN, because AP's position is fixed unlike the MN's. Nonetheless, it is also worth pointing out that sometimes the neighbouring AP might not be captured as *neighbour* due to erratic Wi-Fi RSSI.

ayout 1

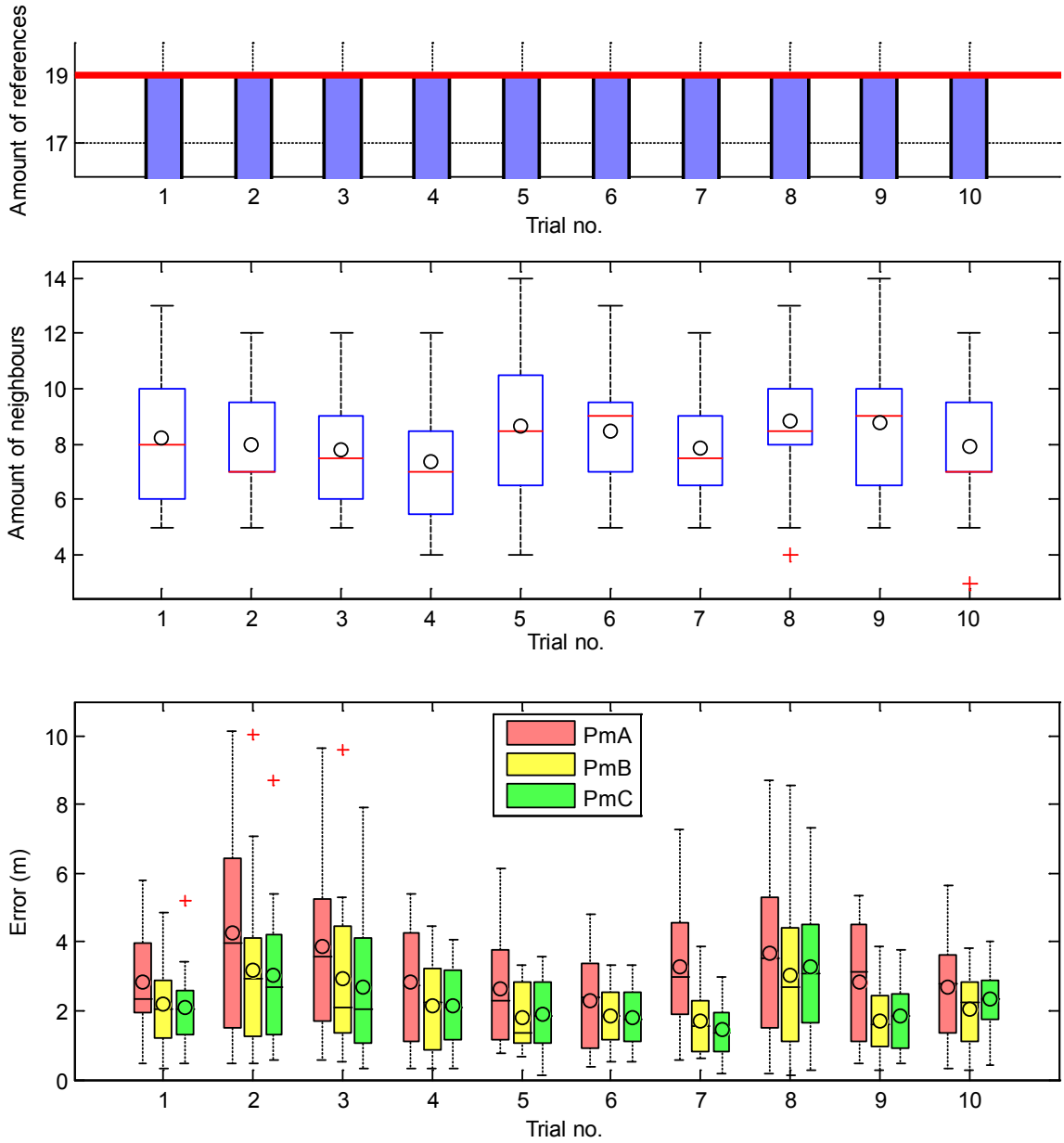


Fig. 5.6 Trials' results for layout no.1.

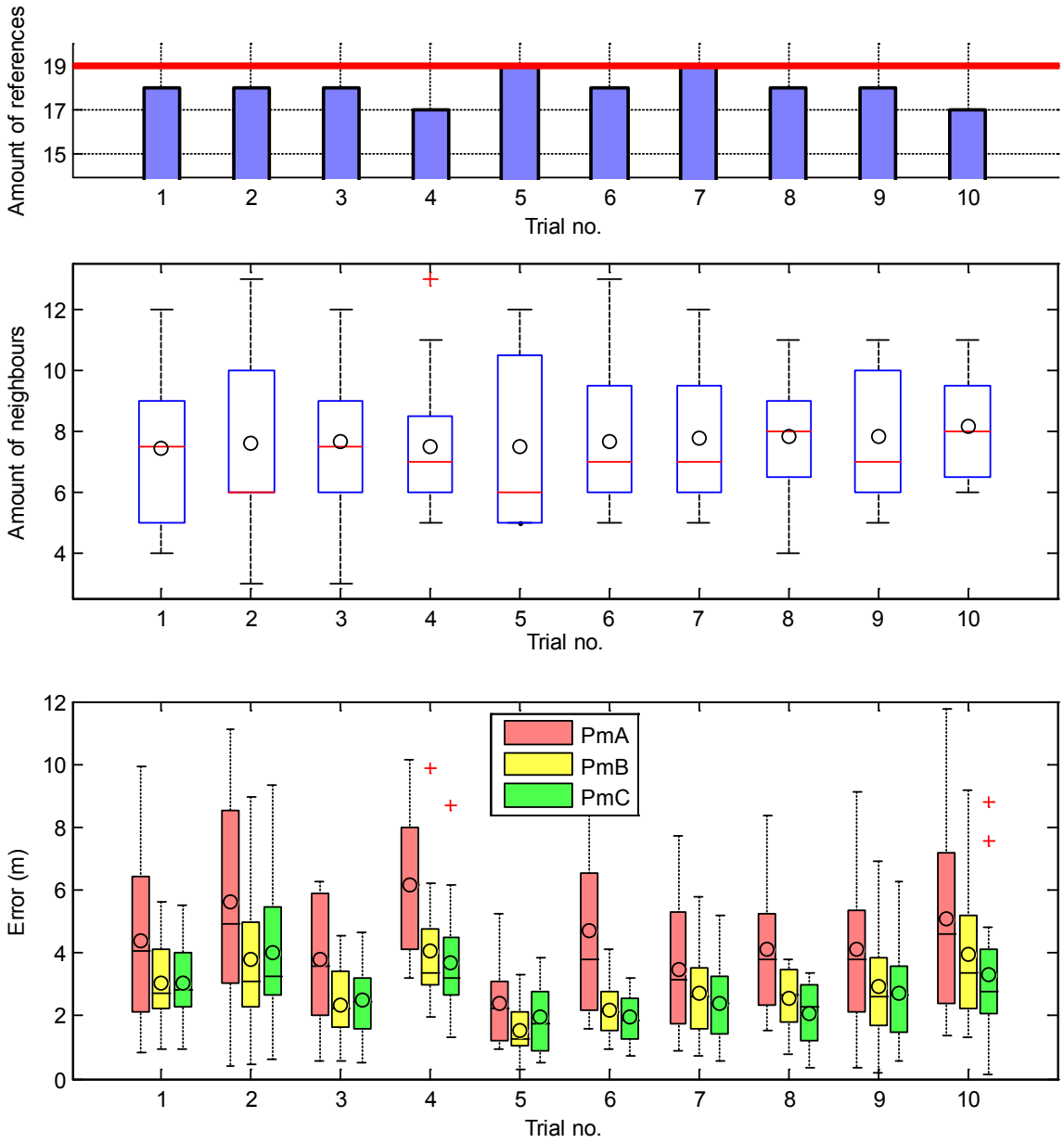


Fig. 5.7 Trials' results for layout no.2.

ayout 3

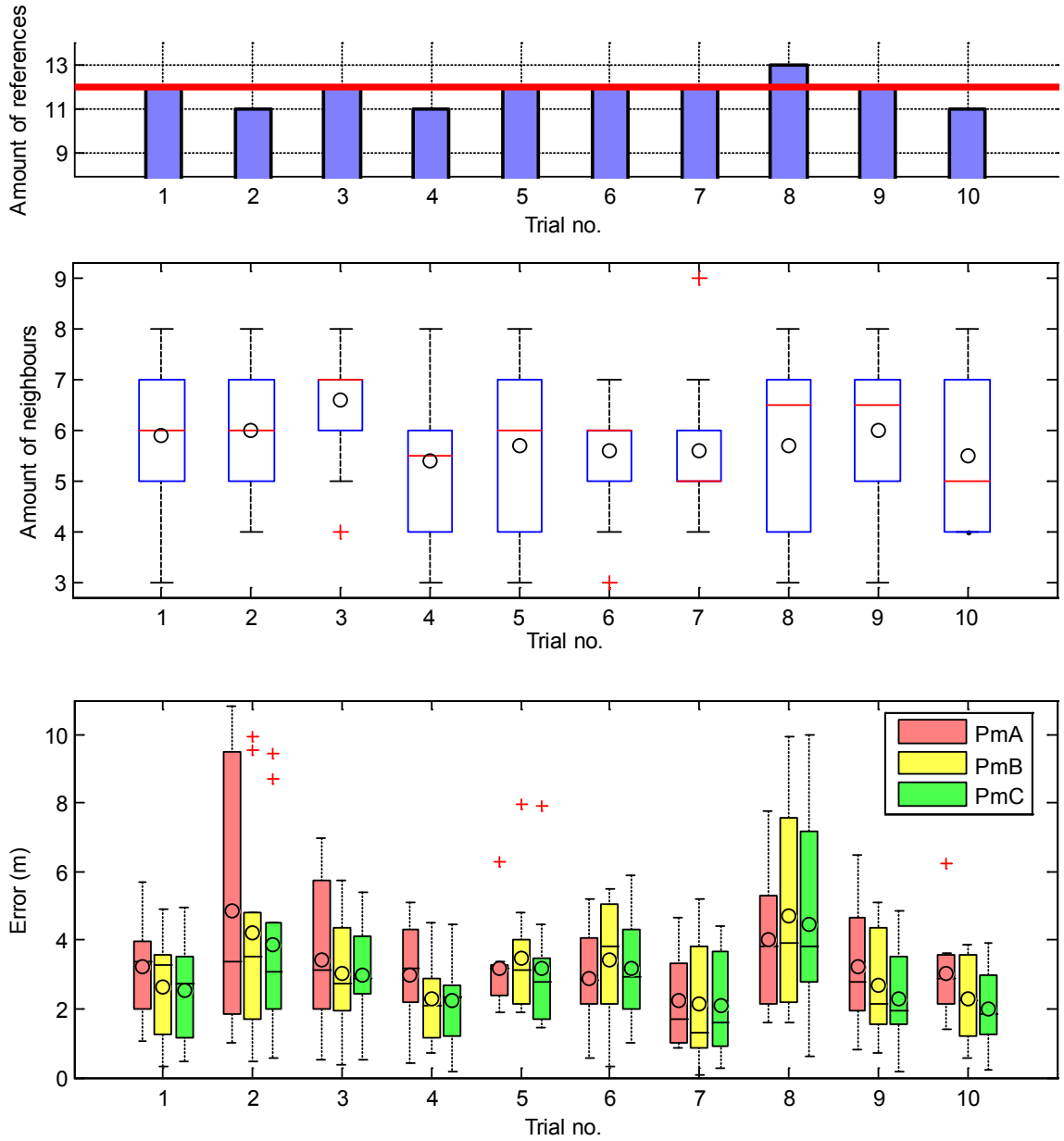


Fig. 5.8 Trials' results for layout no.3.

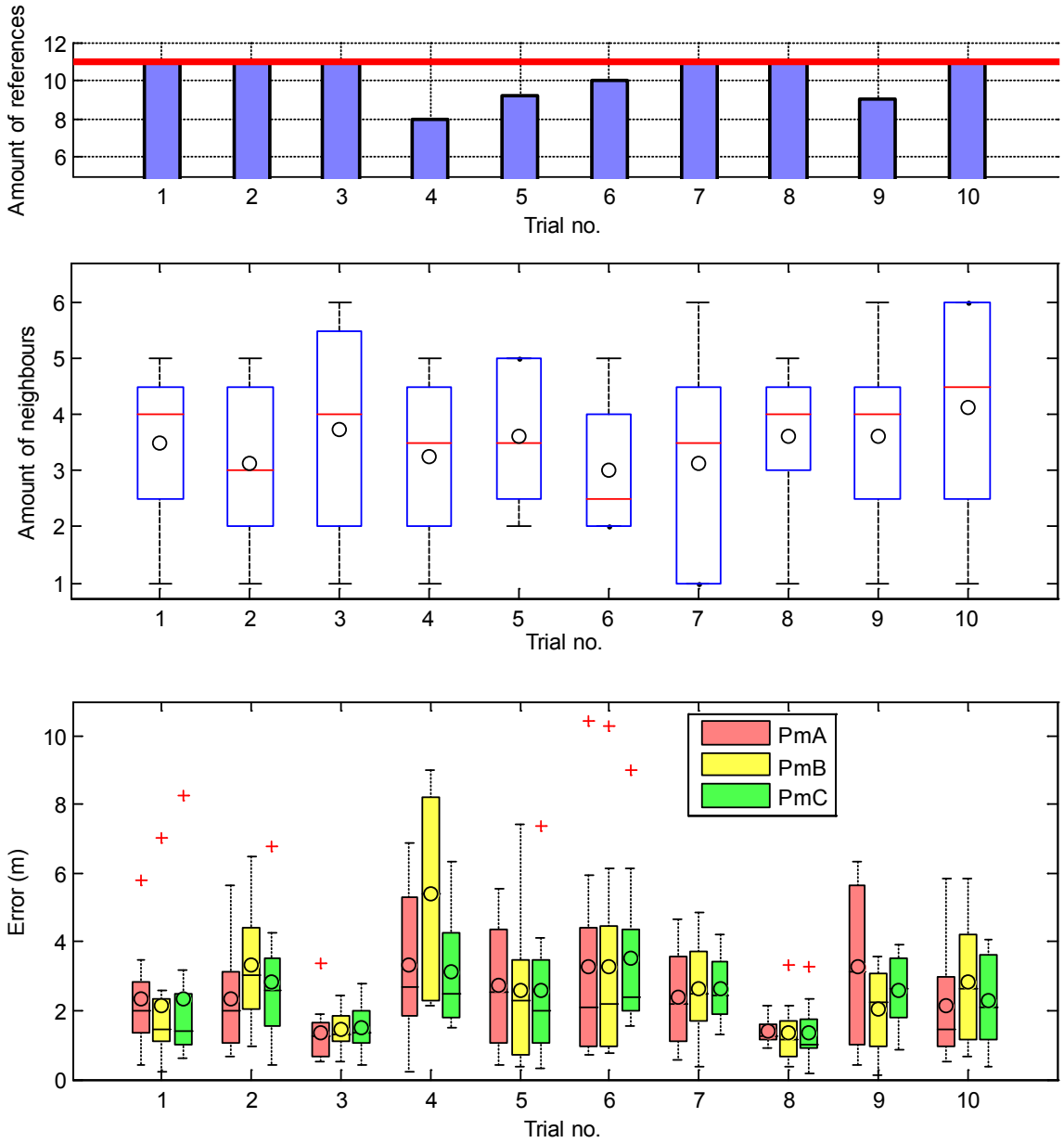


Fig. 5.9 Trials' results for layout no.4.

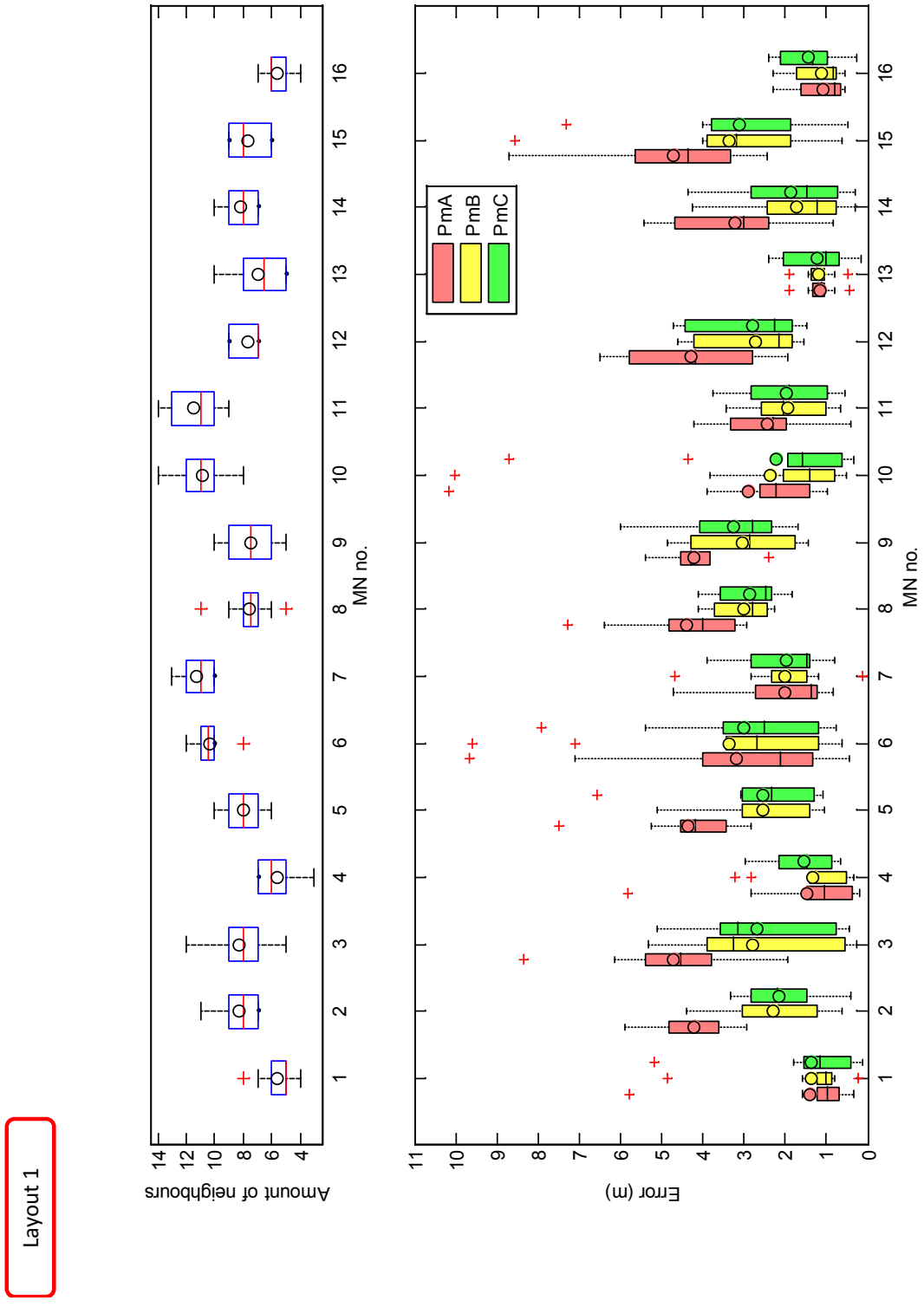


Fig. 5.10 Trials' results for layout no.1 (Re-arranged).

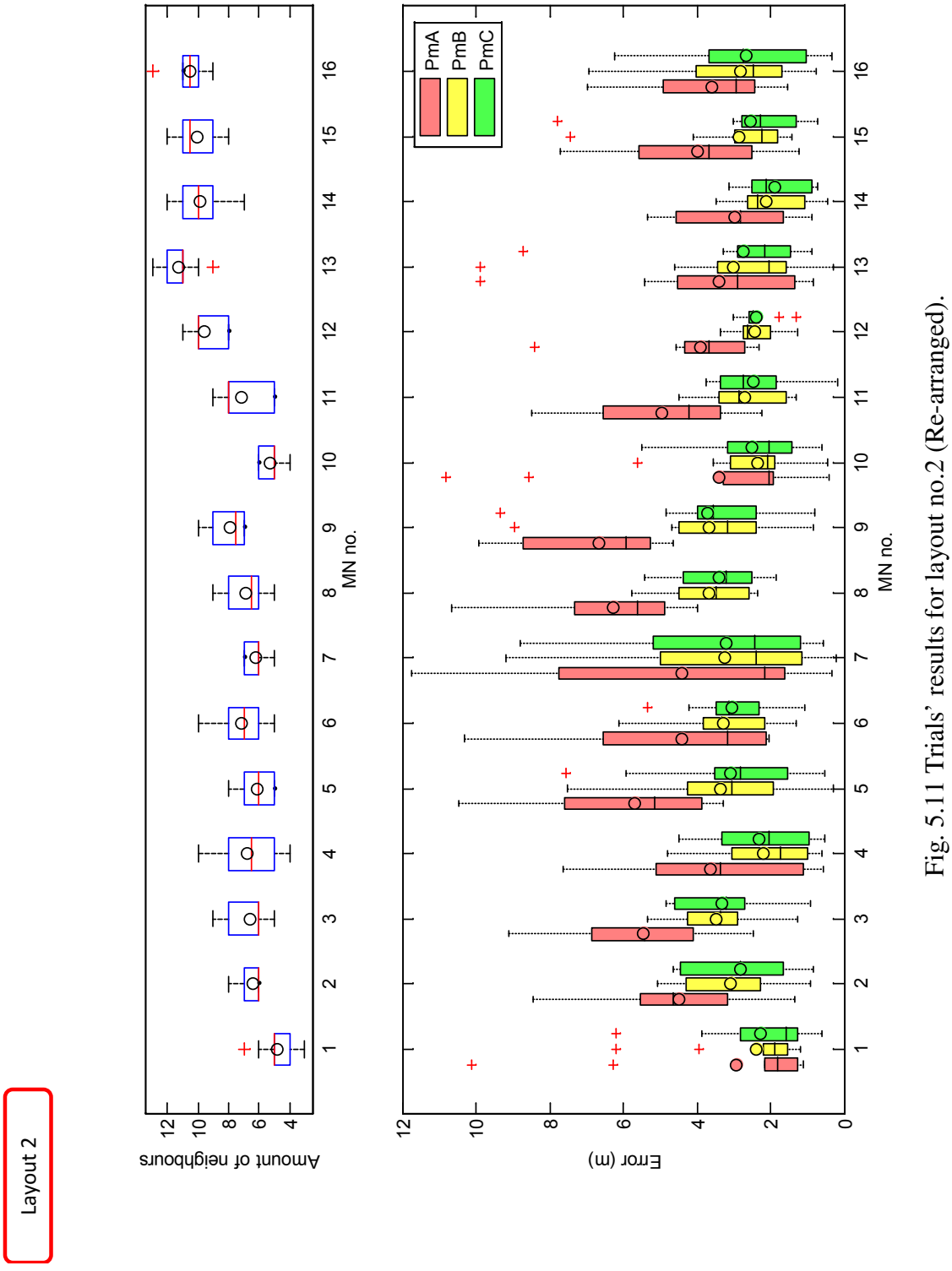


Fig. 5.11 Trials' results for layout no.2 (Re-arranged).

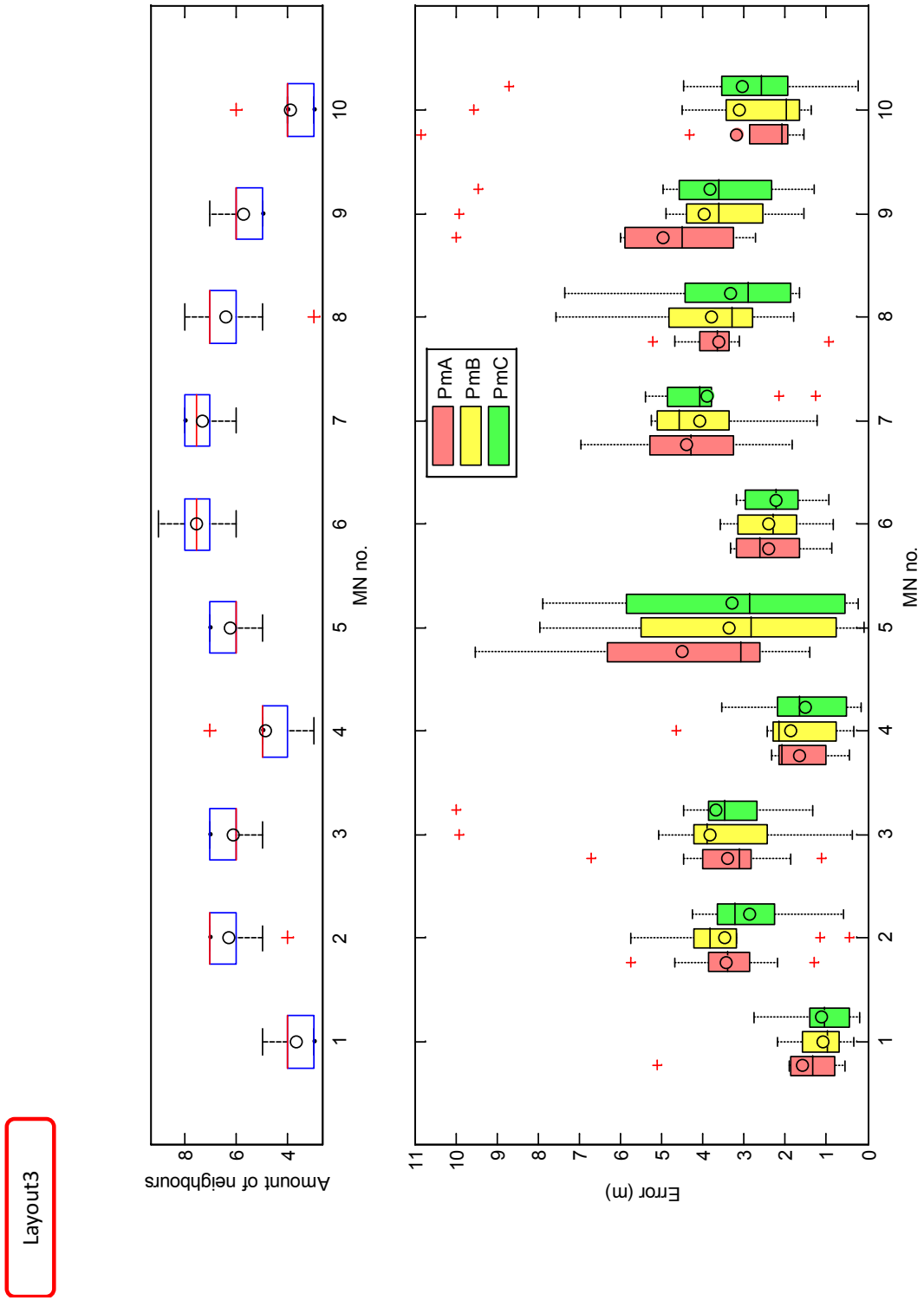


Fig. 5.12 Trials' results for layout no.3 (Re-arranged).

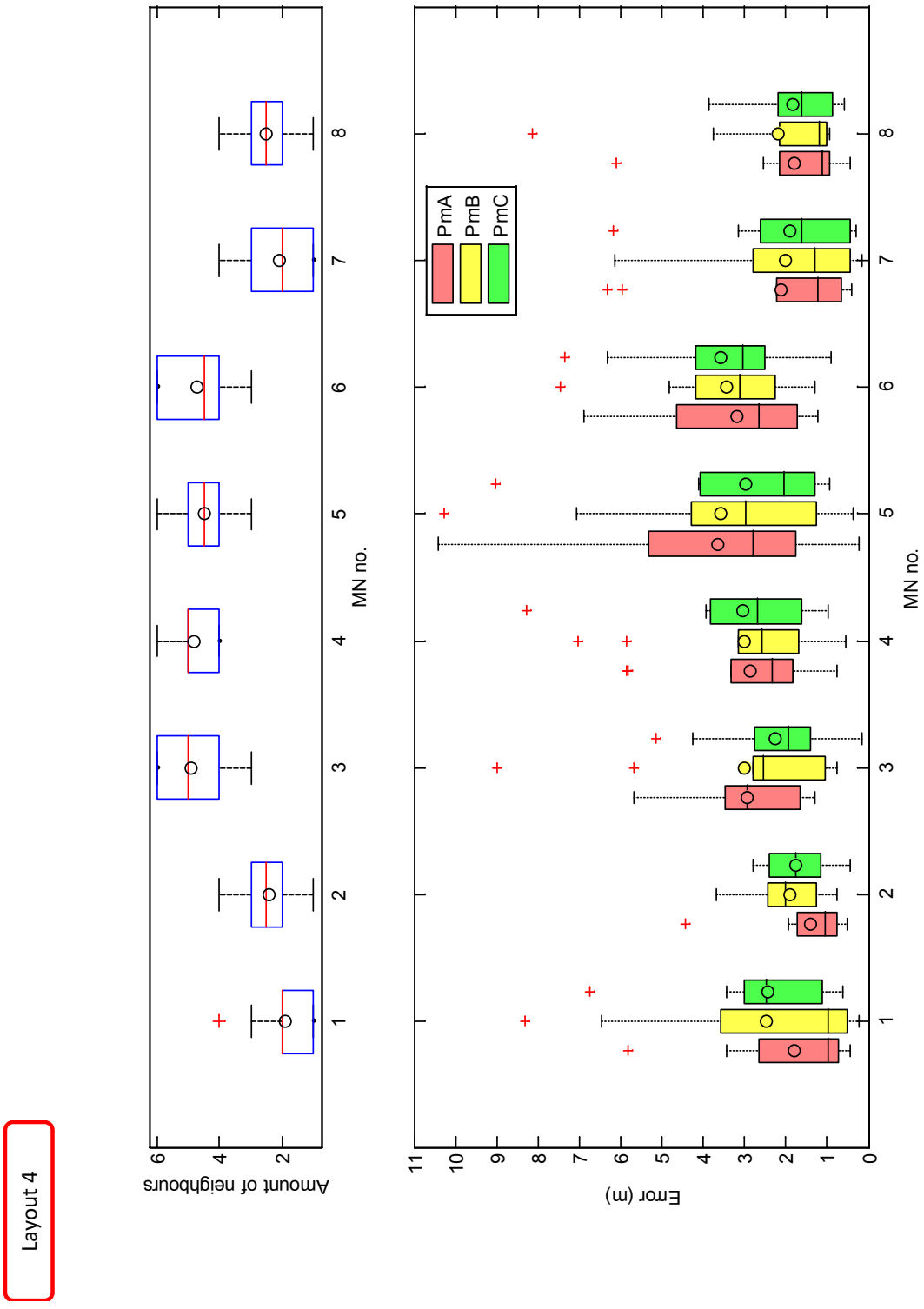


Fig. 5.13 Trials' results for layout no.4 (Re-arranged).

Impact of lesser references on Errors of proposed method

One novel feature of the proposed collaborative method is the usage of directed graph to acquire more references before applying the correction algorithm. The references primarily comprise the neighbouring nodes, and possibly include the neighbouring nodes of neighbouring nodes as well as the distant ones. The proposed correction algorithm then makes use of every available references to refine the estimated position via iterating Multi-Lateration with custom weighted averaging. *PmA*, *PmB* and *PmC* could also be modified in order that only the *neighbours* of respective MNs are used as references, by replacing the line no.3 of **Algorithm 5**, **Algorithm 6** and **Algorithm 7** with "**if** *j*-th node is a *neighbour* of *i*-th node **then**". From Figure 5.14 that compares the errors of original versions to those of modified versions, it is apparent that, except for *PmA*, the original versions outperform the modified versions in all four layouts. Therefore, one can deduce that greater number of references considered during correction is likely to mitigate the errors more effectively.

Impact of varying number of BNs on Errors of proposed method

As previously mentioned, BNs are more influential than MNs while refining the MN's estimated position (P_i, z). To examine the impact of having BNs on the correction's effectiveness, the number of Wi-Fi APs allowed to be included in the graph was varied according to cases shown in Table 5.1. Note that the APs were regarded as BNs, and the number of APs allowed

Table 5.1 Number of APs allowed to be included in the Directed Graph for different cases

Case	APs allowed
1	AP1, AP2, AP3, AP4
2	AP1, AP2, AP3
3	AP1, AP2
4	AP1
5	None

does not always equate to the number of BNs existing in the graph because sometimes the AP may not be captured as *neighbour* to any MNs even though they are indeed very nearby. Figure 5.15 compares the errors of all three proposed methods in all five cases for all four layouts. Here, the accuracy metric is the upper quartile of the boxplot. The lower the upper quartile, the higher the accuracy. From Figure 5.15, one can observe that: firstly, the accuracy of all three proposed methods deteriorates as the number of APs allowed decreases; secondly, highest accuracy's deterioration rate goes to *PmA*, followed by *PmB* and then *PmC*; thirdly,

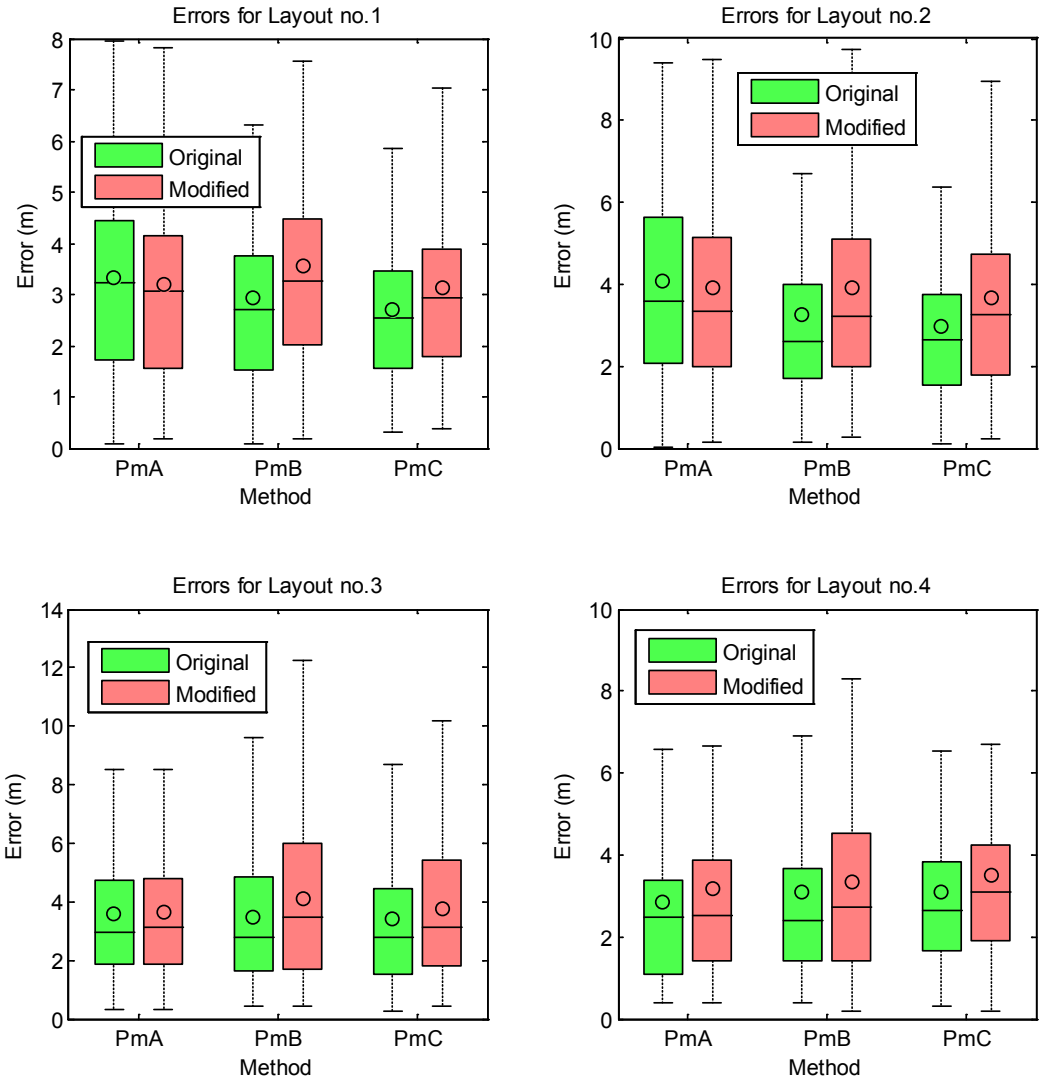


Fig. 5.14 Errors of proposed methods: All references versus Neighbours only.

among the three, *PmC* has the highest accuracy in all cases except case 5 for layout no.4. Figure 5.16 shows one of the trials' results in case 5 where BN is absent from the graph. From Figure 5.16, it is apparent that the positions estimated by *PmC* visually form a *distribution* that considerably matches the MNs' actual positions, yet the *distribution* is slightly turned clockwise and swayed to the right. The *distribution* could be shifted westward if 'AP4' were a *neighbour* of 4-th MN. This is because the estimated position of 4-th MN might be shifted towards 'AP4', thereby also somewhat "pulling" the estimated positions of other MNs towards the same directions. Generally, in whichever direction the *distribution* sways to, a BN needs to be available at opposite direction to mitigate the sway.

Discussion

By comparing the errors (upper quartiles) of *PmA*, *PmB* and *PmC* shown in Figure 5.15 with the errors (upper quartiles) of *TriB* shown in Figure 5.5, one can observe that: firstly, *PmA* seemingly outperforms *TriB* only in cases 1 & 2 for all four layouts; secondly, *TriB* outperforms *PmB* only in cases 4 & 5 for layout no.3 as well as in case 5 for layout no.4; thirdly, *PmC* outperforms *TriB* in all cases for any layouts except case 5 for layout no.4. Such statistics implies two things: firstly, *PmA*, *PmB* and *PmC* are deemed pointless in the aforementioned cases because *TriB* outperforms them despite being comparatively much simpler in terms of computation/algorithm; secondly, among the three proposed methods, *PmC* is the most robust towards the lack of BNs.

Admittedly, Beacon Node is not always available at the right place to anchor the MNs' estimated positions, and therefore alternative way of anchoring is necessary. One potential way to mitigate the sway of *distribution* is shifting the whole *distribution* until its centroid matches with the centroid of actual positions. An illustration for such shifting is depicted in Figure 5.17. Note that the estimated positions and actual positions in reality are unlikely to be so neatly arranged. Figure 5.18, Figure 5.19, Figure 5.20 and Figure 5.21 depict the errors of proposed methods before and after shifting. From these four Figures, it is apparent that, in any cases and layouts, the errors after the shifting reduce and also become more consistent. However, the main problem with such shifting is locating the actual centroid while the actual positions of all MNs are originally unknown. Nevertheless, the actual centroid could be possibly found if we know how the MNs are distributed within a known area, e.g. if the people (MNs) are somewhat evenly scattered within a hall, then the actual centroid is simply assumed the centroid of the hall.

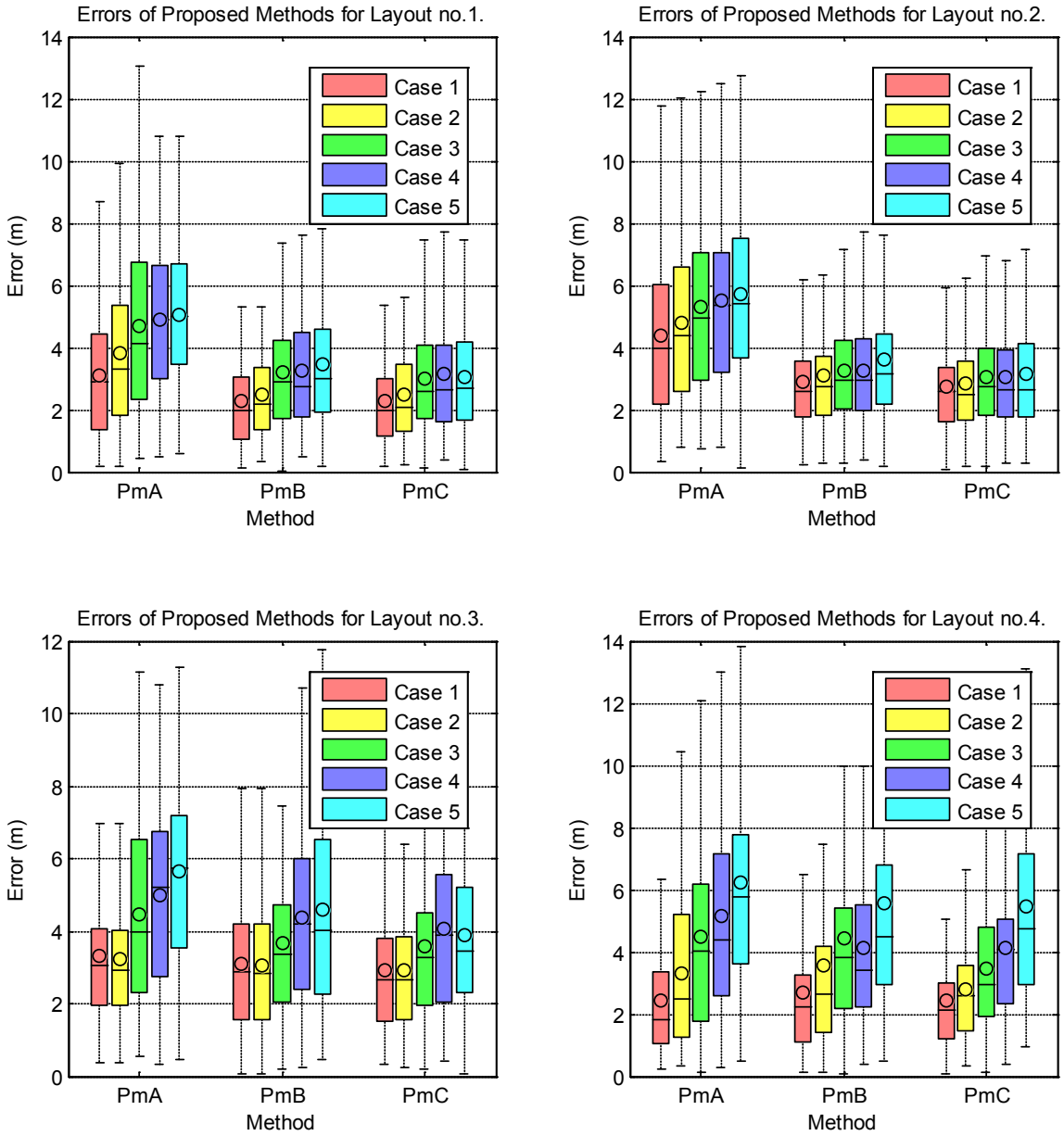


Fig. 5.15 Errors in all five cases for all four layouts.

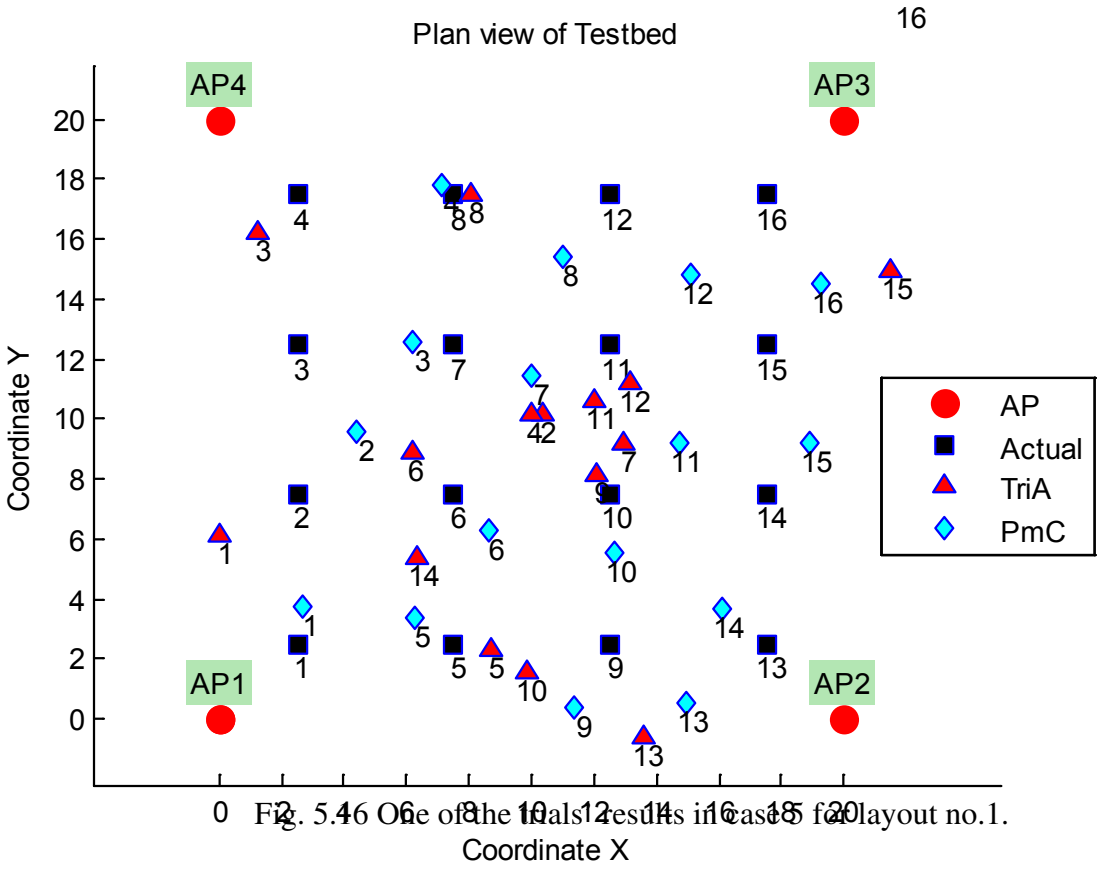


Fig. 5.16 One of the trials results in case 5 for layout no.1.

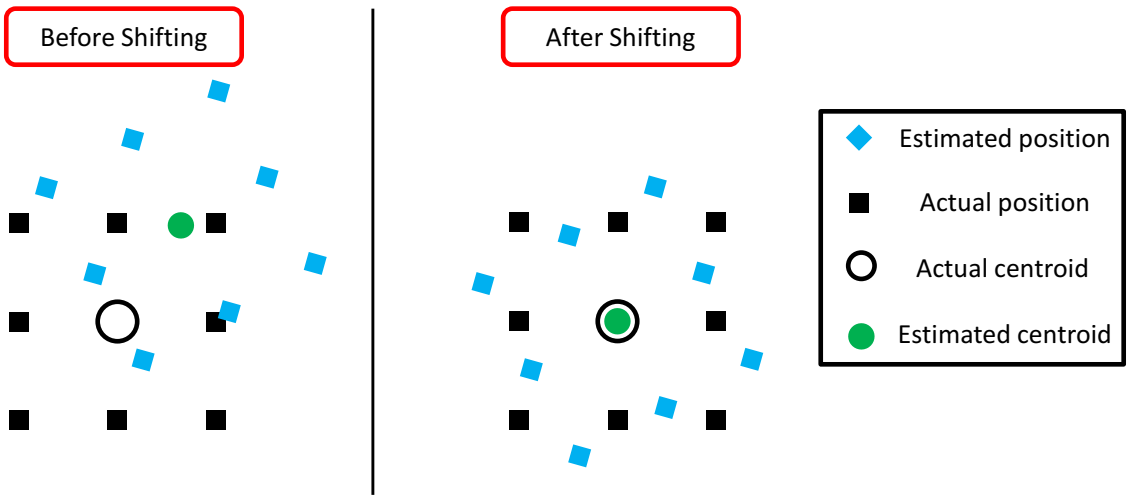


Fig. 5.17 Illustration: Estimated positions before and after shifting based on centroid.

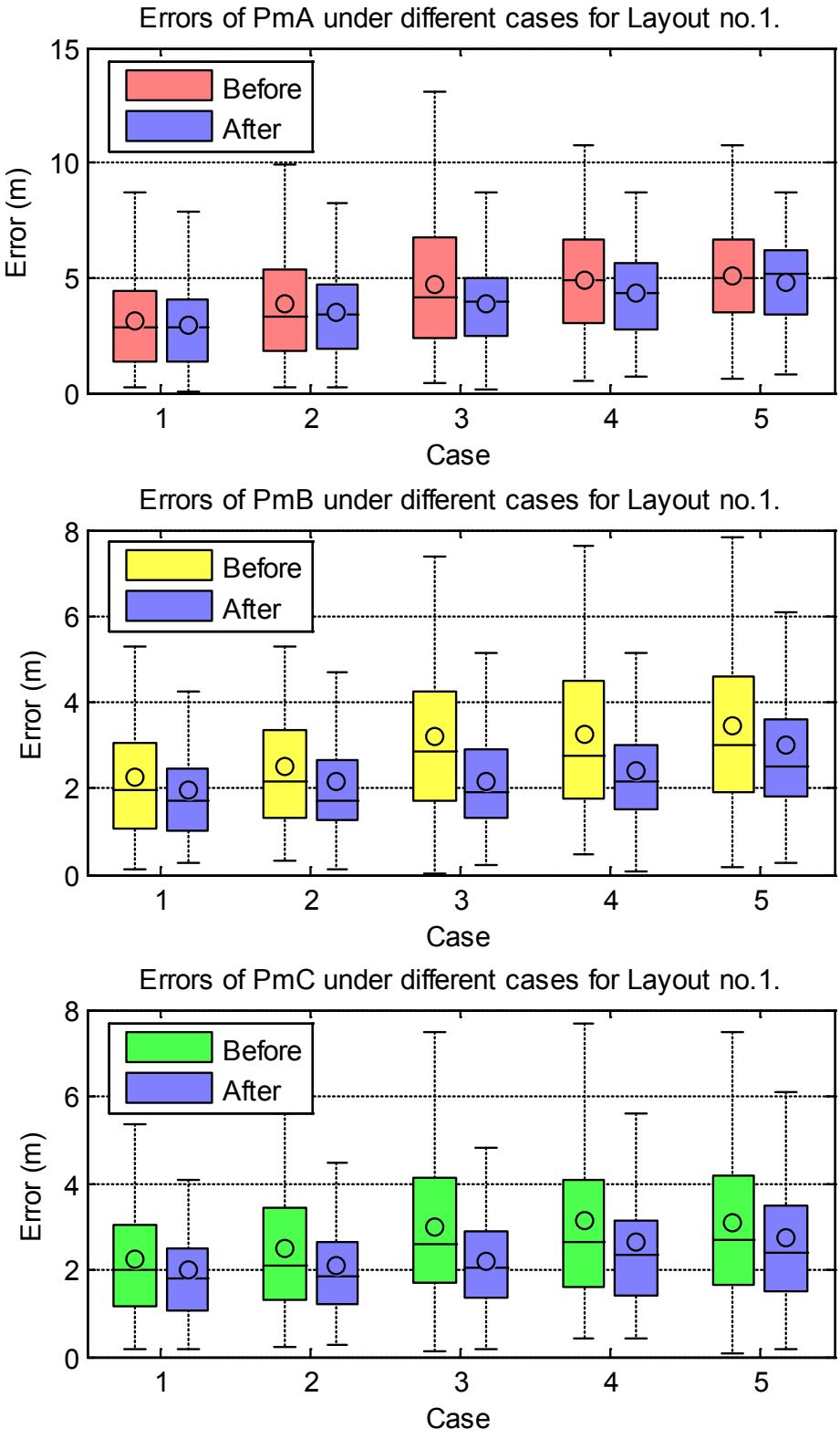


Fig. 5.18 Errors of proposed methods Before and After shifting (Layout no.1).

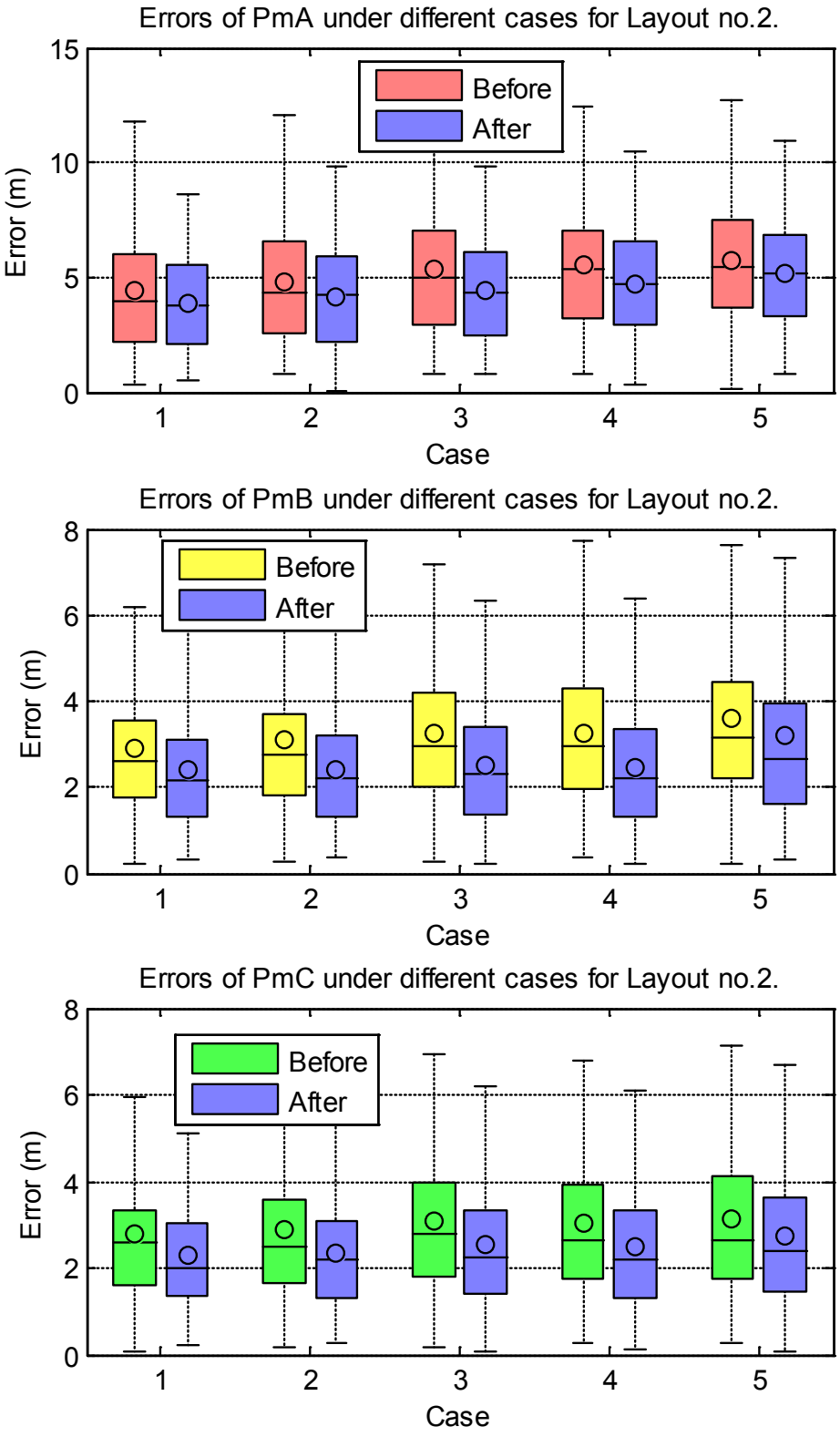


Fig. 5.19 Errors of proposed methods Before and After shifting (Layout no.2).

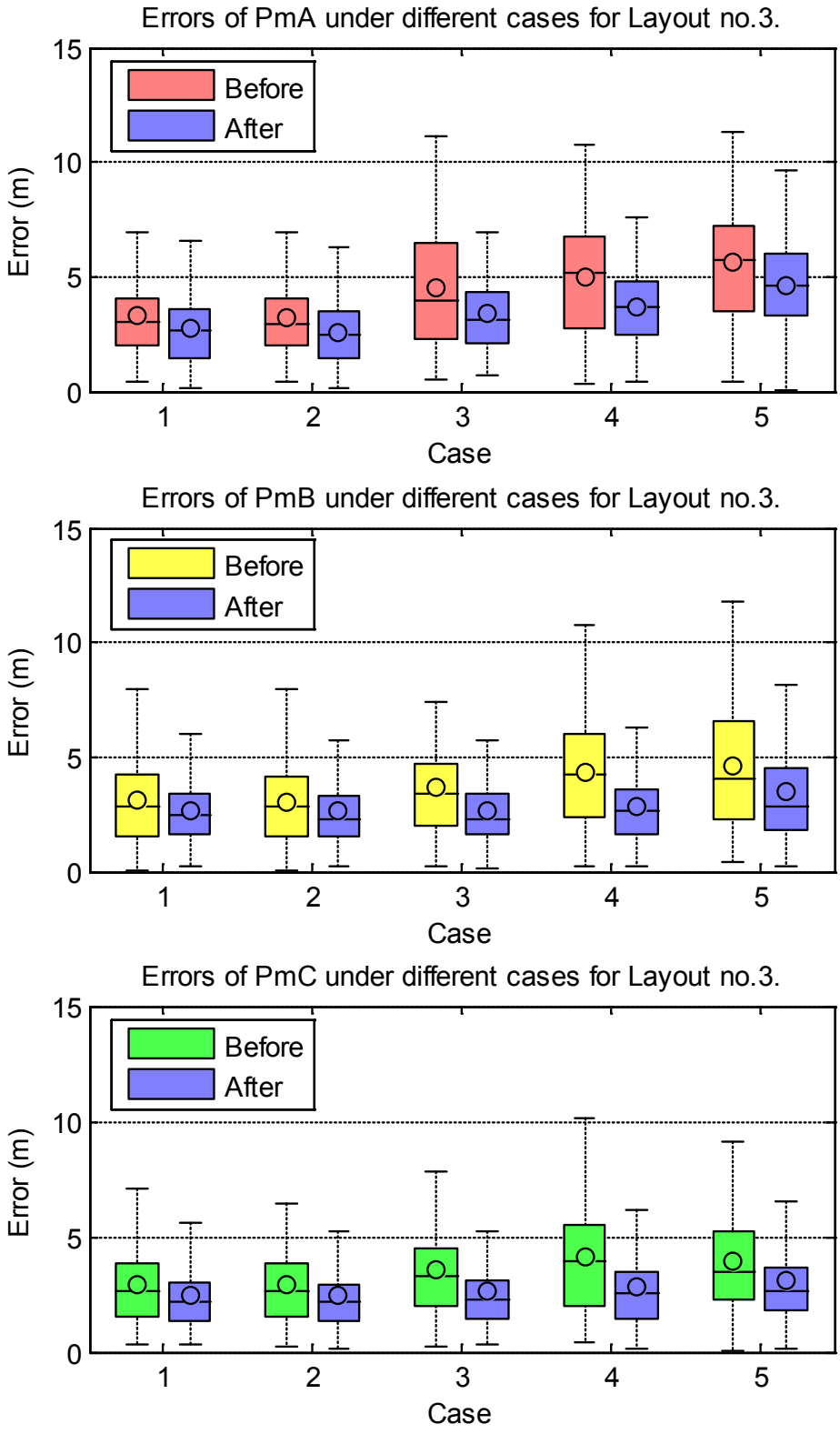


Fig. 5.20 Errors of proposed methods Before and After shifting (Layout no.3).

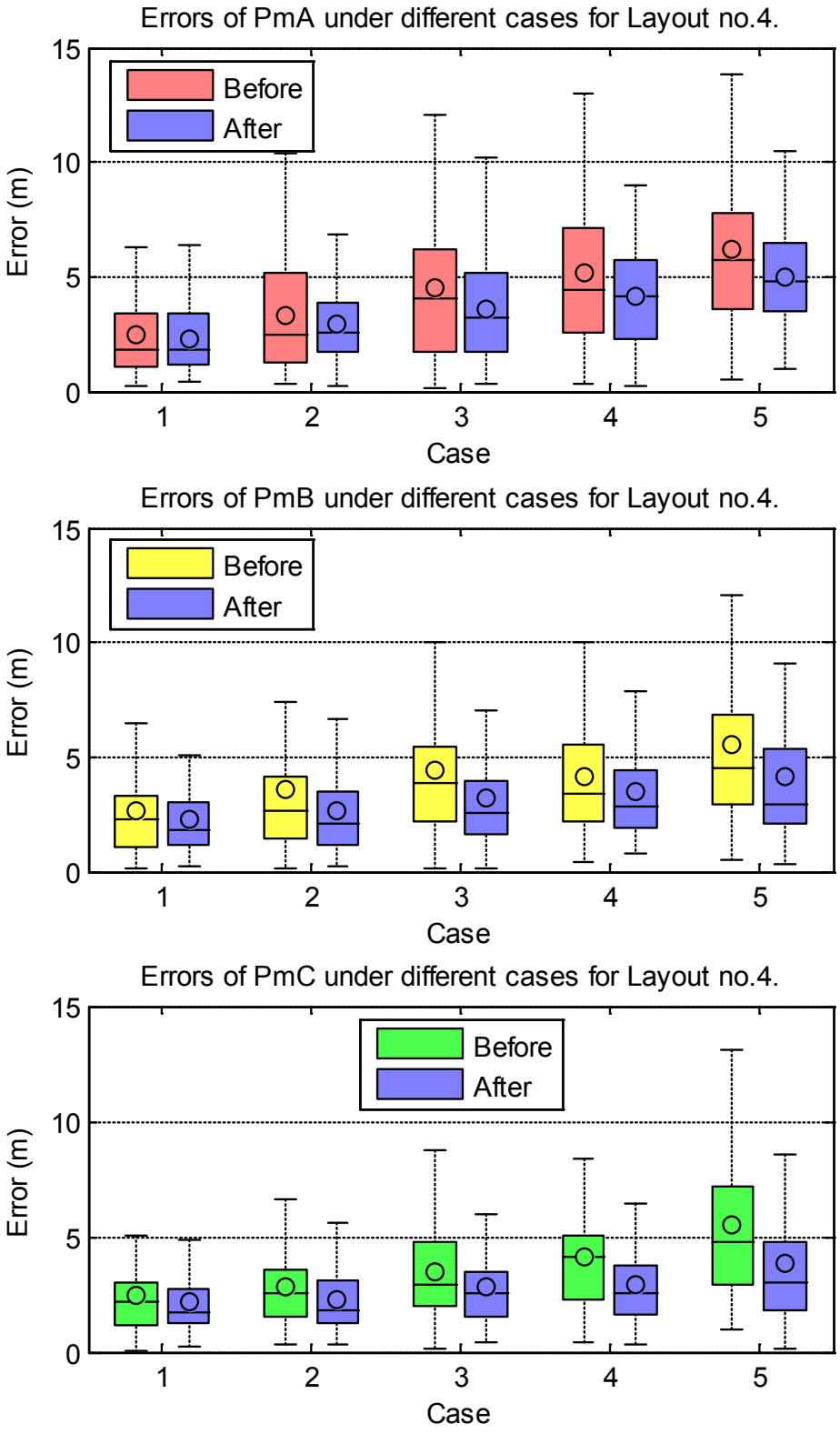


Fig. 5.21 Errors of proposed methods Before and After shifting (Layout no.4).

5.4 Evaluation II

Figure 5.22 shows the plane view of the testbed used to evaluate the efficacy of proposed collaborative method in improving the positions estimated by some PDR -based positioning methods. Note that the four Wi-Fi APs at the site were just deployed to provide RSSI data, and they were not utilized as BNs by the proposed collaborative method under this testbed.

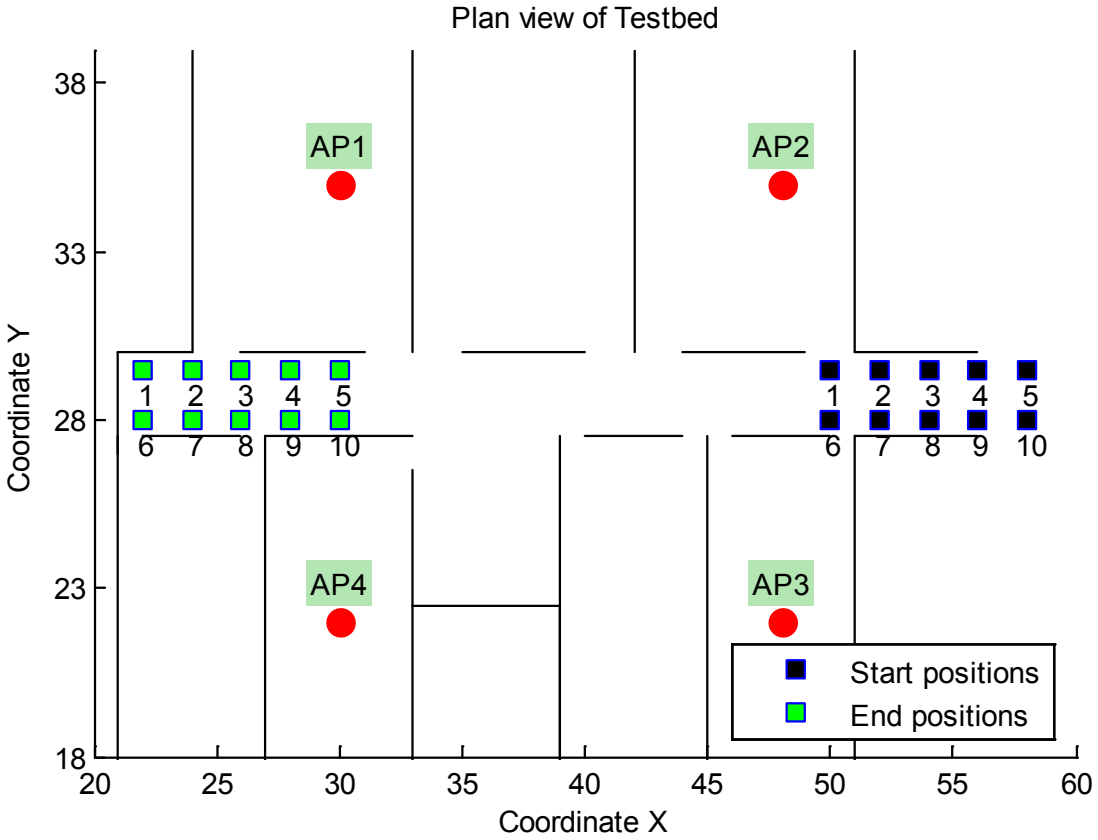


Fig. 5.22 Plane view of testbed.

A total of ten trials were conducted. In each trial, ten MNs walked a long straight path from their respective start positions until the end positions, and their positions were estimated by five different PDR based methods which are listed as follows:

1. *PdrA*: Sole PDR, as explained in **Chapter 3.3**.
2. *PdrB*: PDR with Markov Chain Monte Carlo sampling, explained in **Chapter 3.4.3**.
3. *PdrC*: PDR with Sampling Importance Resampling, explained in **Chapter 3.4.2**.
4. *PdrD*: PDR with heading correction, as explained in **Chapter 4.2**.

The initial headings of all MNs were provided. However, all MNs were required to make few random turnings at their start positions before commencing the walk, and the resultant headings were estimated by using Equation (3.4). The random turnings before the walk were intended to introduce some heading errors in order to simulate unreliable PDR. Note that PDR is well known for being vulnerable to erroneous heading. Then, when all MNs reached their respective end positions, the proposed collaborative method was executed to correct the end positions estimated by the aforementioned PDR based methods.

Besides, another collaborative method was also applied to correct the position estimates given by aforementioned PDR based methods, and the corresponding results were compared with those of proposed collaborative method. This method, which is adapted from [81], corrects the estimated position of each MN via a Kalman Filter based algorithm summarized by **Algorithm 8**. In **Algorithm 8**, t denotes the time instance, P_{t-1} is MN's estimated position provided by *PdrA*, *PdrB*, *PdrC* or *PdrD*; both A and H are 2×2 identity matrices; Q and R are the measurement noise covariance and process noise covariance respectively, which are 2×2 identity matrices multiplied with some appropriate scalars; and the error covariance $\mathbf{P}$ is initially zero.

Algorithm 8 Collaborative method adapted from [81]

1: Predict next state: $x \leftarrow A \times P_{t-1}$
2: Predict next covariance: $\mathbf{P}_t \leftarrow A \times \mathbf{P}_{t-1} \times A^\intercal + Q$
3: Compute Kalman gain: $K \leftarrow \mathbf{P}_t \times H^\intercal \left(H \times \mathbf{P}_t \times H^\intercal + R\right)$
4: **if** This MN has any *neighbours* **then**
5: Obtain measurement: $z_t \leftarrow$ average of estimated positions of all *neighbours*
6: Update predicted state: $x \leftarrow x + K \times (z_t - H \times x)$
7: Update predicted covariance: $\mathbf{P}_t \leftarrow \mathbf{P}_t - K \times H \times \mathbf{P}_t$
8: **end if**
9: Finalize estimated position: $P_t \leftarrow x$

The error metric is the absolute difference between MN's actual end position and estimated end position. Figure 5.23 compares the errors of all aforementioned PDR based methods before and after being refined by the collaborative method, and each boxplot show the distribution of $N_T \times N_m$ errors where N_t is the number of trials and N_m is the number of MNs. Note that PM denotes the proposed collaborative method while EM denotes the collaborative method described by **Algorithm 8**.

From Figure 5.23, it is apparent that PM outperforms EM in correcting the estimated end positions of *PdrA*, *PdrB* and *PdrD* respectively. Besides, the errors of both *PdrA* and

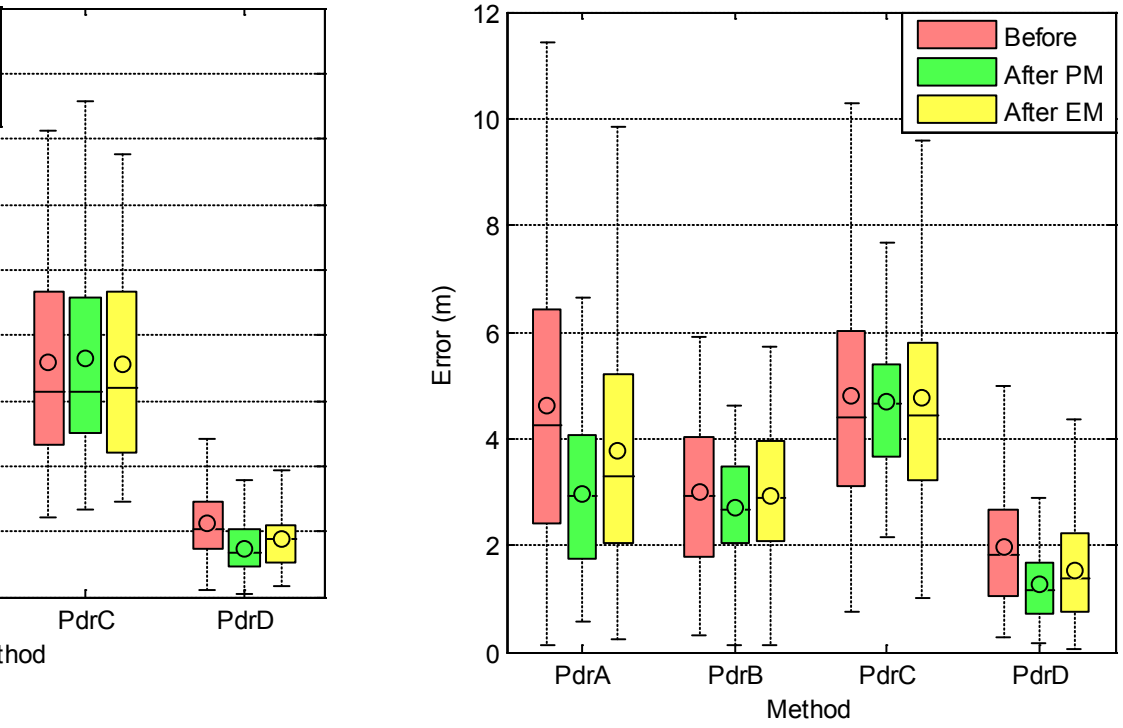


Fig. 5.23 Errors of four different PDR based methods: Before and After correction by collaborative method.

PdrC before correction are similar, yet differ considerably after correction; the reason behind this can be comprehended by referring to a trial's results shown in Figure 5.24. From Figure 5.24, one can observe that: firstly, the PM-corrected estimates are not as spread out as the non-corrected estimates, in other words, the separation distances among the estimates are somewhat improved after correction; secondly, the centroid of non-corrected estimates somewhat determines the centroid of corrected estimates; thirdly, PM-corrected estimates of both *PdrA* and *PdrD*, as compared to those of *PdrB* and *PdrC*, are distributed relatively closer to the actual positions. This therefore deduces that the correction by PM is likely to be more effective when the centroid of non-corrected estimates is nearer to the centroid of actual positions, though the separation distances among non-corrected estimates might be overly large.

Basically, PM corrects the estimated positions by adjusting the separation distances among them, and the resultant estimates do not necessarily become closer to their respective actual positions. For example, referring to the *PdrB*-estimates shown in Figure 5.23, the correction causes the estimated position of MN no.10 to shift further apart from its actual position, while significantly improves its separation distances from estimated positions of multiple MNs (i.e. MN no.1, MN no.4, MN no.5, MN no.6, MN no.7, and MN no.8).

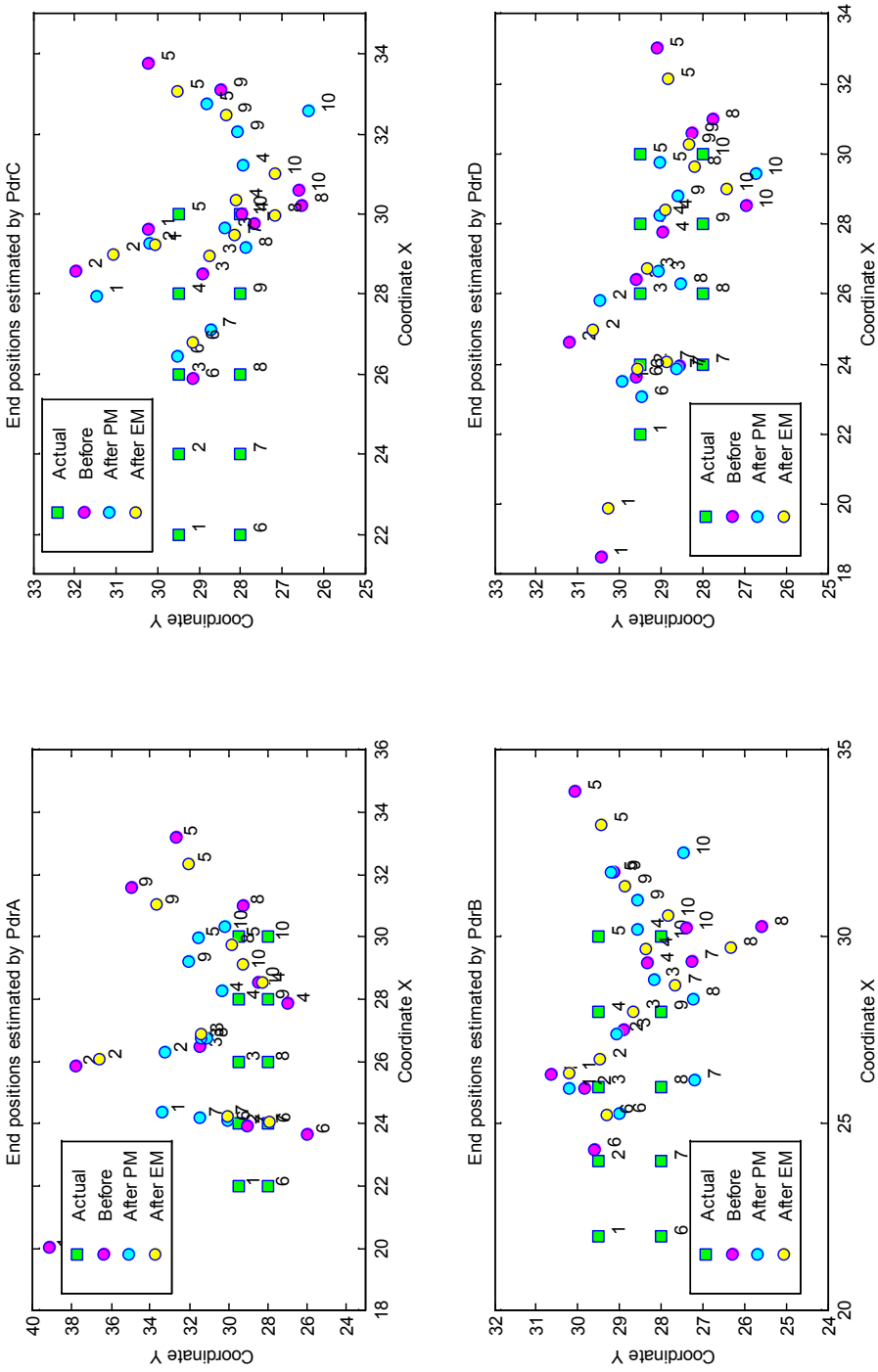


Fig. 5.24 End positions estimated by four different PDR based methods (in first trial).

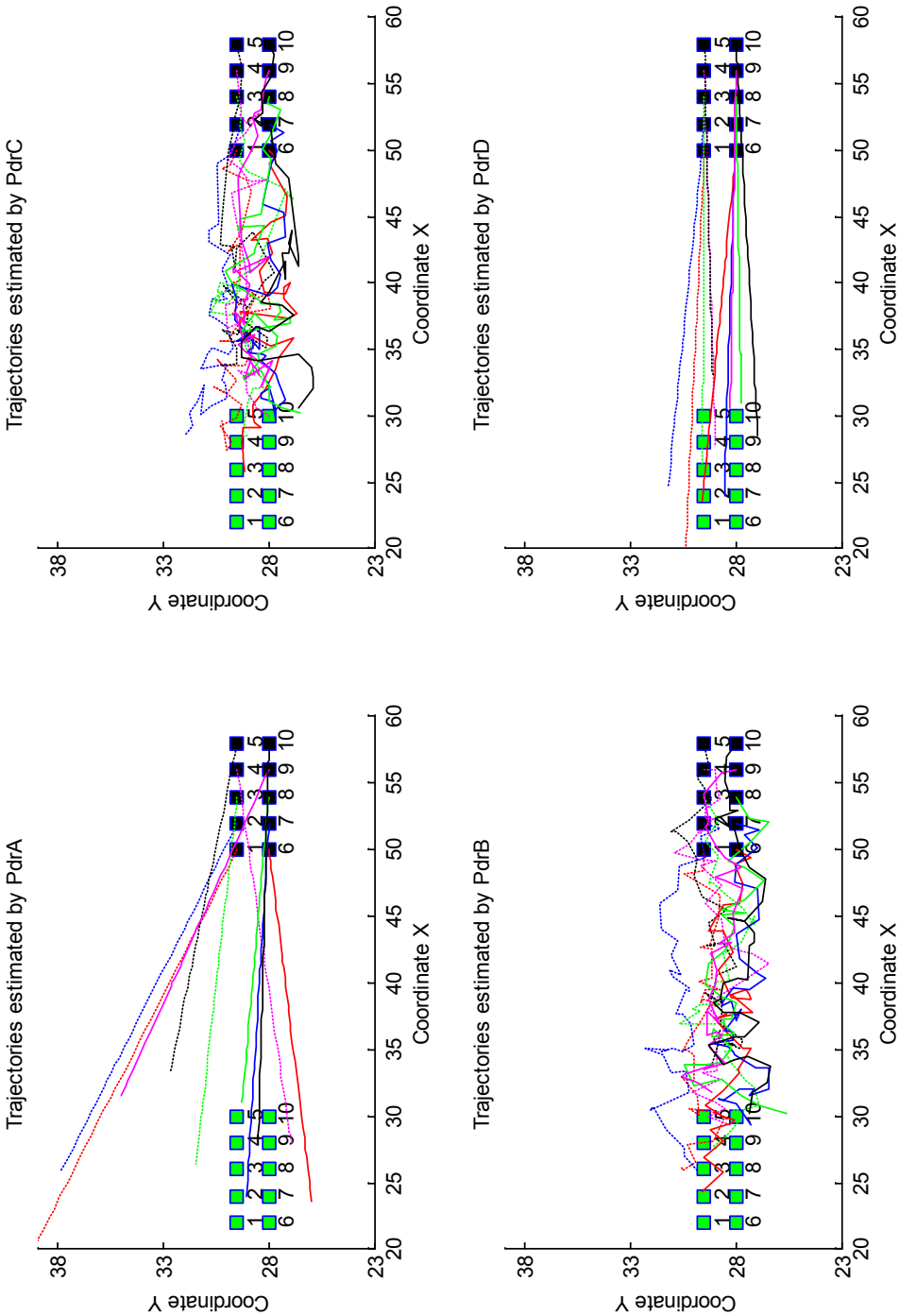


Fig. 5.25 Trajectories estimated by four different PDR based methods (in first trial).

5.5 Summary

This chapter proposes a collaborative positioning method that is based on Directed Graph. The positions of all MNs can be initially estimated via either PDR or RSSI-based Tri-Lateration. The directed graph is used to express the relations among the nodes (both MNs and BNs). From the graph, the shortest distances from each MN to all other nodes may be derived. The derived shortest distances, are then utilized to correct the estimated position of each MN via an iterative process. In every iteration of correction process, estimated position of each MN is finalized by averaging the sample points. The sample points are generated based on the derived shortest distances via a Particle Filter. Among the sample points, only those which are resulted from neighbouring BNs hold comparatively greater weights. The proposed collaborative positioning method had been tested via simulation. The test results are presented to demonstrate the feasibility and efficacy of the proposed method. The accuracy of proposed method is likely to improve when BNs are available at the right spots to mitigate the skewness of finalized estimates.

Chapter 6

Conclusion

6.1 Summary

In view of arising demand for indoor location based services (LBS), many indoor positioning systems (IPS) have been developed by leveraging various technologies such as Micro-Electro-Mechanical Systems (MEMS), Radio Frequency Identification (RFID). However, none of these existing IPS are gaining mainstream adoption as much as their outdoor counterpart i.e. GPS, probably owing to their corresponding implementation cost. The implementation cost of an IPS includes the resources required for system's start-up and maintenance, as well as constraints that might be imposed to the users. Perhaps the most promising IPS are those which leverage the smart-devices (i.e. smartphones and tablets), because smart-devices are increasingly widespread and they are richly equipped with capabilities like GPS, Wi-Fi, Bluetooth and motion detection. With such capabilities of smart-device, various sorts of positioning algorithms are made possible and their underlying techniques are typically Pedestrian Dead-Reckoning (PDR), Lateration and Fingerprinting.

PDR estimates the pedestrian's present position by advancing their last known position with heading and distance which are derived from data of motion sensors, i.e. accelerometers, gyroscopes and magnetometers. PDR is well known for its relatively high positioning accuracy which deteriorates over time due to inherent sensors' biases and drifts. Unlike PDR which can be implemented on smart-device alone, Lateration and Fingerprinting require necessary devices to be deployed at the site to enable wireless transmission between devices. Lateration deduces the pedestrian's position based on RSSI-derived distance between pedestrian and reference device, while Fingeprinting infers pedestrian's position by matching online RSSI measurements with pre-collected RSSI measurements. Albeit the latter, as compared to the former, is more reliable and robust towards erratic RSSI data, its implementation

is comparatively more complex. Further discussion regarding the fundamentals of indoor positioning and related work are presented in **Chapter 2**.

In **Chapter 4**, two schemes that improve PDR's long-term positioning accuracy by utilizing the RSSI of Wi-Fi signals are proposed. The implementation of either scheme is deemed low-cost since Wi-Fi APs are prevalent especially among the indoors. Upon detecting strong RSSI value, the first scheme ensures that the estimated position lies within acceptable region around the corresponding Wi-Fi AP. Subsequently, the estimated position can be further refined if the bearing of the pedestrian from that AP is known. The second scheme, on the other hand, corrects the PDR-estimated headings based on linear regression, and subsequently updates the PDR-estimated positions. However, it is triggered only upon detecting a long straight PDR-estimated trajectory. Nevertheless, such trajectory is commonly observed in pedestrian's daily movements such as walking down the long corridors within the university or mega-mall. Both schemes had been proven feasible at real sites within the Swinburne University of Technology Sarawak Campus, and experimental results show that PDR with proposed schemes outperform some existing PDR -based positioning methods in terms of accuracy and robustness.

More information taken into consideration is likely to result in more reliable estimation, and thus positioning methods that estimate individual pedestrian's position by utilizing surrounding pedestrians' information are gaining interest among IPS researchers. Such collaborative methods are promising due to the fact that the pedestrians usually cluster either knowingly or unknowingly among themselves, and most of them carry pervasive-computing devices such as smart-devices and laptops. Therefore, in **Chapter 5**, a novel collaborative positioning method that makes use of directed graph is proposed. The directed graph is constructed to express the relations among the pedestrians and site's anchors/APs. The pedestrians' positions are initially estimated via either PDR or RSSI-based Tri-Lateration, and then refined by a Particle Filter based algorithm according to the information derived from the graph. Both experimental and simulation results suggest the feasibility and efficacy of the proposed method.

6.2 Future Research Work

There are several issues that can be explored to further refine the methods proposed in this thesis, and a number of them are listed as follows:

1. As mentioned in **Chapter 4.2.1**, all elements of *series*, regardless of whether they are outliers, are indiscriminately considered in deriving the linear regression line. The impact of outliers on the orientation of linear regression line should be investigated as in which and how many outliers among the *series*' elements can be useful or negligible. Subsequently, weighted linear regression may also be attempted.

2. During the heading correction, the original headings (estimated by PDR) are simply replaced by the headings which are derived based on linear regression. Admittedly, the derived headings are not always more accurate than the original headings. Perhaps a potential way to achieve more reliable heading correction is by combining the original headings with the derived headings via some kinds of filters. Alternatively, indoor floor plan (if provided) may be useful in determining whether heading correction is necessary – for example, if the original long straight trajectory does not cross any boundaries, then heading correction is not needed. Furthermore, the re-estimated trajectory (which is resulted from corrected headings) should also reasonably align with the floor plan [44, 57, 61].

3. As mentioned in **Chapter 5.2**, the neighbouring BNs are considered more reliable references than other nodes while adjusting the estimated position of individual pedestrian. However, BNs (i.e. Wi-Fi APs) might not be in near vicinity of pedestrian clusters at all times, and therefore alternative source of reliable references is needed. In such cases, pedestrians may be considered as reliable references if they fulfill certain criteria - for example, being stationary for a period of time and moving in the same groups. In addition, the weight of each reference may be assigned according to the criteria it fulfills.

4. As described in **Chapter 5.4**, the proposed collaborative method was executed only when all MNs reached their respective actual end positions. In other words, only the PDR-estimated end positions experience the correction. It was intended under the presumption that, by the time the MN arrives at the end of route, the position error would reach a stage where correction is necessary. At short intervals, the PDR accuracy is relatively high, and so the correction may seem unnecessary. Therefore, the right timing for the correction should be investigated.

5. For the proposed collaborative method to work effectively, all MNs are required to monitor RSSI from surrounding nodes continuously. Such requirement may cause the proposed collaborative method prohibitively complex and energy heavy besides compromising on privacy issues. This, however, is not discussed in this work because the proposed collaborative method was designed so merely as an effort to examine the possibility of achieving reliable indoor positioning by making use of the collaboration among pervasive computing devices.

6. The proposed schemes are heavily reliant on empirically defined parameters and thresholds which are likely to differ across different device models. Different devices may observe dissimilar RSSI values even at same position. Moreover, they may vary in terms of scan response time, sensor readings, etc.

References

[1] Abadi, M. J., Luceri, L., Hassan, M., Chou, C. T., and Nicoli, M. (2014). A collaborative approach to heading estimation for smartphone-based pdr indoor localisation. In *Indoor Positioning and Indoor Navigation (IPIN), 2014 International Conference on*, pages 554–563.

[2] Abdulrahim, K., Moore, T., Hide, C., and Hill, C. (2014). Understanding the performance of zero velocity updates in mems-based pedestrian navigation. *International Journal of Advancements in Technology*, 5(2):53–60.

[3] Ahmed, S. H., Bouk, S. H., Javaid, N., and Sasase, I. (2012). Combined human, antenna orientation in elevation direction and ground effect on rssi in wireless sensor networks. In *Frontiers of Information Technology (FIT), 2012 10th International Conference on*, pages 46–49.

[4] Alarifi, A., Al-Salman, A., Alsaleh, M., Alnafessah, A., Al-Hadhrami, S., Al-Ammar, M. A., and Al-Khalifa, H. S. (2016). Ultra wideband indoor positioning technologies: Analysis and recent advances. *Sensors*, 16(5).

[5] Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *Signal Processing, IEEE Transactions on*, 50(2):174–188.

[6] AT&T Laboratories Cambridge (2005 [cited 19 August 2016]). The bat ultrasonic location system. Available from: http://www.cl.cam.ac.uk/research/dtg/attarchive/bat/.

[7] Au, A. W. S., Feng, C., Valaee, S., Reyes, S., Sorour, S., Markowitz, S. N., Gold, D., Gordon, K., and Eizenman, M. (2013). Indoor tracking and navigation using received signal strength and compressive sensing on a mobile device. *IEEE Transactions on Mobile Computing*, 12(10):2050–2062.

[8] Bertram, J. and Ruina, A. (2001). Multiple walking speed-frequency relations are predicted by constrained optimization. *Journal of Theoretical Biology*, 209:445–453.

[9] Bolic, M., Rostamian, M., and Djuric, P. M. (2015). Proximity detection with rfid: A step toward the internet of things. *IEEE Pervasive Computing*, 14(2):70–76.

[10] Carrillo, D., Moreno, V., Úbeda, B., and Skarmeta, A. F. (2015). Magicfinger: 3d magnetic fingerprints for indoor location. *Sensors*, 15(7).

[11] Chan, L.-w., Chiang, J.-r., Chen, Y.-c., Ke, C.-n., Hsu, J., and Chu, H.-h. (2006). Collaborative localization: Enhancing wifi-based position estimation with neighborhood links in clusters. In *Proceedings of the 4th International Conference on Pervasive Computing*, PERVASIVE'06, pages 50–66, Berlin, Heidelberg. Springer-Verlag.

[12] Chen, Z. (2003). Bayesian filtering: From Kalman Filters to Particle Filters, and Beyond.

[13] Cho, S. Y. and Park, C. G. (2006). Mems based pedestrian navigation system. *Journal of Navigation*, 59:135–153.

[14] Constandache, I., Choudhury, R. R., and Rhee, I. (2010). Towards mobile phone localization without war-driving. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9.

[15] Correa, A., Barcelo, M., Morell, A., and Lopez Vicario, J. (2014). Enhanced inertial-aided indoor tracking system for wireless sensor networks: A review. *Sensors Journal, IEEE*, 14(9):2921–2929.

[16] Dardari, D., Closas, P., and Djuric, P. (2015). Indoor tracking: Theory, methods, and technologies. *Vehicular Technology, IEEE Transactions on*, 64(4):1263–1278.

[17] Dias, M. H. C. and Siqueira, G. L. (2005). Indoor tdoa-aoa measurements at the 3g systems frequency band – a simple approach. *Journal of Communication and Information Systems*, 20(3):112–124.

[18] Fang, L., Antsaklis, P., Montestruque, L., McMickell, M., Lemmon, M., Sun, Y., Fang, H., Koutroulis, I., Haenggi, M., Xie, M., and Xie, X. (2005). Design of a wireless assisted pedestrian dead reckoning system - the navmote experience. *IEEE Transactions on Instrumentation and Measurement*, 54(6):2342–2358.

[19] Foxlin, E. (2005). Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Computer Graphics and Applications*, 25(6):38–46.

[20] Graefenstein, J., Albert, A., Biber, P., and Schilling, A. (2009). Wireless node localization based on rssi using a rotating antenna on a mobile robot. In *Positioning, Navigation and Communication, 2009. WPNC 2009. 6th Workshop on*, pages 253–259.

[21] Graefenstein, J. and Bouzouraa, M. E. (2008). Robust method for outdoor localization of a mobile robot using received signal strength in low power wireless networks. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 33–38.

[22] Gu, Y., Lo, A., and Niemegeers, I. (2009). A survey of indoor positioning systems for wireless personal networks. *IEEE Communications Surveys Tutorials*, 11(1):13–32.

[23] Gusenbauer, D., Isert, C., and Krösche, J. (2010). Self-contained indoor positioning on off-the-shelf mobile devices. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, pages 1–9.

[24] Harle, R. (2013). A survey of indoor inertial positioning systems for pedestrians. *IEEE Communications Surveys Tutorials*, 15(3):1281–1293.

[25] He, S. and Chan, S. H. G. (2016). Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys Tutorials*, 18(1):466–490.

[26] He, Y. and Bilgic, A. (2011). Iterative least squares method for global positioning system. *Advances in Radio Science*, 9:203–208.

[27] Hightower, J. and Borriello, G. (2001). Location systems for ubiquitous computing. *Computer*, 34(8):57–66.

[28] Higuchi, T., Fujii, S., Yamaguchi, H., and Higashino, T. (2014). Mobile node localization focusing on stop-and-go behavior of indoor pedestrians. *IEEE Transactions on Mobile Computing*, 13(7):1564–1578.

[29] Higuchi, T., Yamaguchi, H., and Higashino, T. (2012). Clearing a crowd: Context-supported neighbor positioning for people-centric navigation. In *Proceedings of the 10th International Conference on Pervasive Computing*, Pervasive'12, pages 325–342, Berlin, Heidelberg. Springer-Verlag.

[30] Honkavirta, V., Perala, T., Ali-Loytty, S., and Piche, R. (2009). A comparative survey of wlan location fingerprinting methods. In *Positioning, Navigation and Communication, 2009. WPNC 2009. 6th Workshop on*, pages 243–251.

[31] Iwase, T. and Shibasaki, R. (2013). Infra-free indoor positioning using only smartphone sensors. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8.

[32] J. Kim, H. Jang, D. H. and Park, C. (2004). A step, stride and heading determination for the pedestrian navigation system. *Journal of Global Positioning Systems*, 3(1-2):273–279.

[33] Jimenez, A., Seco, F., Prieto, C., and Guevara, J. (2009). A comparison of pedestrian dead-reckoning algorithms using a low-cost mems imu. In *IEEE International Symposium on Intelligent Signal Processing. WISP 2009.*, pages 37–42.

[34] Jin, Y., Motani, M., Soh, W. S., and Zhang, J. (2010). Sparsetrack: Enhancing indoor pedestrian tracking with sparse infrastructure support. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9.

[35] Jin, Y., Soh, W.-S., Motani, M., and Wong, W.-C. (2013). A robust indoor pedestrian tracking system with sparse infrastructure support. *Mobile Computing, IEEE Transactions on*, 12(7):1392–1403.

[36] Jin, Y., Toh, H.-S., Soh, W.-S., and Wong, W.-C. (2011). A robust dead-reckoning pedestrian tracking system with low cost sensors. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 222–230.

[37] Jo, C. and Lee, C. (2016). Multilateration method based on the variance of estimated distance in range-free localisation. *Electronics Letters*, 52(12):1078–1080.

[38] Kalman, D. (2002). An underdetermined linear system for gps. *The College Mathematics Journal*, 33:384–390.

[39] Khmou, Y., Safi, S., and Frikel, M. (2014). Comparative Study between Several Direction of Arrival Estimation Methods. *Journal of Telecommunications and Information Technology*, 1:41–48.

[40] Klingbeil, L. and Wark, T. (2008). A wireless sensor network for real-time indoor localisation and motion monitoring. In *Information Processing in Sensor Networks, 2008. IPSN '08. International Conference on*, pages 39–50.

[41] Lan, K.-C. and Shih, W.-Y. (2014). Using smart-phones and floor plans for indoor location tracking. *IEEE Transactions on Human-Machine Systems*, 44(2):211–221.

[42] Lee, B. G. and Chung, W. Y. (2011). Multitarget three-dimensional indoor navigation on a pda in a wireless sensor network. *IEEE Sensors Journal*, 11(3):799–807.

[43] Levi, R. and Judd, T. (1996). Dead reckoning navigational system using accelerometer to measure foot impacts. US Patent 5,583,776.

[44] Li, F., Zhao, C., Ding, G., Gong, J., Liu, C., and Zhao, F. (2012). A reliable and accurate indoor localization method using phone inertial sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 421–430, New York, NY, USA. ACM.

[45] Liew, L. S. and Wong, W. S. H. (2016). Improved pedestrian dead-reckoning-based indoor positioning by rssi-based heading correction. *IEEE Sensors Journal*, 16(21):7762–7773.

[46] Lingwen Zhang, C. T. and Yang, G. (2011). Wireless positioning: Fundamentals, systems and state of the art signal processing techniques. *Cellular Networks - Positioning, Performance Analysis, Reliability*.

[47] Liu, H., Darabi, H., Banerjee, P., and Liu, J. (2007). Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(6):1067–1080.

[48] Liu, H. F., Ren, W., Zhang, T., Gong, J., m. Liang, J., Liu, B., w. Shi, J., and Huang, Z. (2014). An adaptive selection algorithm of threshold value in zero velocity updating for personal navigation system. In *Control Conference (CCC), 2014 33rd Chinese*, pages 1035–1038.

[49] Liu, Y., Wang, Q., Liu, J., and Wark, T. (2012). Mcmc-based indoor localization with a smart phone and sparse wifi access points. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, pages 247–252.

[50] Luo, M. (2013). *Indoor radio propagation modeling for system performance prediction*. PhD dissertation, INSA de Lyon.

[51] Luo, Y., Hoeber, O., and Chen, Y. (2013). Enhancing wi-fi fingerprinting for indoor positioning using human-centric collaborative feedback. *Human-centric Computing and Information Sciences*, 3(1):1–23.

[52] Mahfouz, S., Mourad-Chehade, F., Honeine, P., Farah, J., and Snoussi, H. (2016). Non-parametric and semi-parametric rssi/distance modeling for target tracking in wireless sensor networks. *IEEE Sensors Journal*, 16(7):2115–2126.

[53] Mardeni, R. and Nizam, O. S. (2010). Node positioning in zigbee network using trilateration method based on the received signal strength indicator (rssi). *European Journal of Scientific Research*, 46(1):48–61.

[54] Müller, P., Raitoharju, M., Ali-Löytty, S., Wirola, L., and Piché, R. (2016). A survey of parametric fingerprint-positioning methods. *Gyroscopy and Navigation*, 7(2):107–127.

[55] Nikitin, P. V., Martinez, R., Ramamurthy, S., Leland, H., Spiess, G., and Rao, K. V. S. (2010). Phase based spatial identification of uhf rfid tags. In *2010 IEEE International Conference on RFID (IEEE RFID 2010)*, pages 102–109.

[56] Noh, Y., Yamaguchi, H., Lee, U., Vij, P., Joy, J., and Gerla, M. (2013). Clips: Infrastructure-free collaborative indoor positioning scheme for time-critical team operations. In *Pervasive Computing and Communications (PerCom), 2013 IEEE International Conference on*, pages 172–178.

[57] Pai, D., Malpani, M., Sasi, I., Aggarwal, N., and Mantripragada, P. (2012). Padati: A robust pedestrian dead reckoning system on smartphones. In *IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 2000–2007.

[58] Payal, A., Rai, C. S., and Reddy, B. V. R. (2015). Experimental analysis of some radio propagation models for smart wireless sensor networks applications. In *SAI Intelligent Systems Conference (IntelliSys), 2015*, pages 338–342.

[59] Pierlot, V. and Droogenbroeck, M. V. (2014). A new three object triangulation algorithm for mobile robot positioning. *IEEE Transactions on Robotics*, 30(3):566–577.

[60] Rafael Berkvens, Dries Vandermeulen, C. V. H. P. and Weyn, M. (2014). Feasibility of geomagnetic localization and geomagnetic ratslam. *International Journal on Advances in Systems and Measurements*, 7(1&2):44–56.

[61] Rai, A., Chintalapudi, K. K., Padmanabhan, V. N., and Sen, R. (2012). Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, pages 293–304, New York, NY, USA. ACM.

[62] Rice, A. and Harle, R. (2005). Evaluating lateration-based positioning algorithms for fine-grained tracking. In *Proceedings of the 2005 Joint Workshop on Foundations of Mobile Computing*, DIALM-POMC '05, pages 54–61, New York, NY, USA. ACM.

[63] Ruizhi, C., Ling, P., and Yuwei, C. (2011). A smart phone based pdr solution for indoor navigation. In *Proceedings of the 24th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2911)*, pages 1404–1408.

[64] S. H. Shin, C. G. Park, H. S. H. and Lee, J. M. (2005). Mems-based personal navigator equipped on the user's body. In *Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2005)*, pages 1998–2002.

[65] Sarkar, T. K., Ji, Z., Kim, K., Medouri, A., and Salazar-Palma, M. (2003). A survey of various propagation models for mobile communication. *IEEE Antennas and Propagation Magazine*, 45(3):51–82.

[66] Schiller, J. and Voisard, A. (2004). *Location Based Services*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[67] Shabbir, N., Sadiq, M. T., Kashif, H., and Ullah, R. (2011). Comparison of radio propagation models for long term evolution (LTE) network. *CoRR*, abs/1110.1519.

[68] Shao, H. J., Zhang, X. P., and Wang, Z. (2014). Efficient closed-form algorithms for aoa based self-localization of sensor nodes using auxiliary variables. *IEEE Transactions on Signal Processing*, 62(10):2580–2594.

[69] Simon, D. (2010). Kalman filtering with state constraints: a survey of linear and nonlinear algorithms. *IET Control Theory Applications*, 4(8):1303–1318.

[70] Smith, A. and Breen, T. (2007). Use of geo-stationary satellites to augment wide_area multilateration synchronization. US Patent App. 11/343,079.

[71] Steinhoff, U. and Schiele, B. (2010). Dead reckoning from the pocket - an experimental study. In *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 162–170.

[72] Stromback, P., Rantakokko, J., Wirkander, S.-L., Alexandersson, M., Fors, K., Skog, I., and Handel, P. (2010). Foot-mounted inertial navigation and cooperative sensor fusion for indoor positioning. In *Proceedings of the 2010 International Technical Meeting of The Institute of Navigation*, pages 89–98.

[73] Susi, M., Renaudin, V., and Lachapelle, G. (2013). Motion mode recognition and step detection algorithms for mobile phone users. *Sensors*, 13(2):1539.

[74] Taniuchi, D., Liu, X., Nakai, D., and Maekawa, T. (2015). Spring model based collaborative indoor position estimation with neighbor mobile devices. *Selected Topics in Signal Processing, IEEE Journal of*, 9(2):268–277.

[75] Wadhwa, M., Song, M., Rali, V., and Shetty, S. (2009). The impact of antenna orientation on wireless sensor network performance. In *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, pages 143–147.

[76] Wang, B., Zhou, S., Liu, W., and Mo, Y. (2015). Indoor localization based on curve fitting and location search using received signal strength. *IEEE Transactions on Industrial Electronics*, 62(1):572–582.

[77] Wang, H., Sen, S., Elgohary, A., Farid, M., Youssef, M., and Choudhury, R. R. (2012). No need to war-drive: Unsupervised indoor localization. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, MobiSys '12, pages 197–210, New York, NY, USA. ACM.

[78] Want, R., Hopper, A., Falcão, V., and Gibbons, J. (1992). The active badge location system. *ACM Trans. Inf. Syst.*, 10(1):91–102.

[79] Weinberg, H. (2002). Using the adxl202 in pedometer and personal navigation applications. *Analog Devices AN-602 Application Note*.

[80] Yang, C. and r. Shao, H. (2015). Wifi-based indoor positioning. *IEEE Communications Magazine*, 53(3):150–157.

[81] Yang, Z., Feng, X., and Zhang, Q. (2014). Adometer: Push the limit of pedestrian indoor localization through cooperation. *Mobile Computing, IEEE Transactions on*, 13(11):2473–2483.

[82] Zahid Farid, R. N. and Ismail, M. (2013). Recent advances in wireless indoor localization techniques and system. *Journal of Computer Networks and Communications*, 2013.

[83] Zhang, R., Hoflinger, F., and Reindl, L. (2013). Inertial sensor based indoor localization and monitoring system for emergency responders. *Sensors Journal, IEEE*, 13(2):838–848.

[84] Zhongliang, D., Yanpei, Y., Xie, Y., Neng, W., and Lei, Y. (2013). Situation and development tendency of indoor positioning. *Communications, China*, 10(3):42–55.

[85] Zhou, S. and Pollard, J. K. (2006). Position measurement using bluetooth. *IEEE Transactions on Consumer Electronics*, 52(2):555–558.