Radio Propagation Modeling and Measurements for ZigBee Based Indoor Wireless Sensor Networks

Danping He, Gabriel Mujica, Guixuan Liang, Jorge Portilla, Teresa Riesgo
Centro de Electronica Industrial
Universidad Politecnica de Madrid
Jose Gutierrez Abascal, 2, 28006 Madrid, Spain
Email:{danping.he, gabriel.mujica, guixuan.liang, jorge.portilla, teresa.riesgo}@upm.es

***Abstract*—The deployment of nodes in Wireless Sensor Networks (WSNs) arises as one of the biggest challenges of this field, which involves in distributing a large number of embedded systems to fulfill a specific application. The connectivity of WSNs is difficult to estimate due to the irregularity of the physical environment and affects the WSN designers' decision on deploying sensor nodes. Therefore, in this paper, a new method is proposed to enhance the efficiency and accuracy on ZigBee propagation simulation in indoor environments. The method consists of two steps: automatic 3D indoor reconstruction and 3D ray-tracing based radio simulation. The automatic 3D indoor reconstruction employs unattended image classification algorithm and image vectorization algorithm to build the environment database accurately, which also significantly reduces time and efforts spent on non-radio propagation issue. The 3D ray tracing is developed by using kd-tree space division algorithm and a modified polar sweep algorithm, which accelerates the searching of rays over the entire space. Signal propagation model is proposed for the ray tracing engine by considering both the materials of obstacles and the impact of positions along the ray path of radio. Three different WSN deployments are realized in the indoor environment of an office and the results are verified to be accurate. Experimental results also indicate that the proposed method is efficient in pre-simulation strategy and 3D ray searching scheme and is suitable for different indoor environments.**

Keywords—ZigBee propagation, 3D ray tracing, environment reconstruction, simulation, measurement.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) are composed of sensor nodes which have the ability to detect surrounding environments and are able to communicate with each other through specific wireless protocols such as ZigBee, Bluetooth and Ultra-wideband (UWB). WSNs can be used in many applications like environment monitoring, security surveillance and home automation. There are numerous algorithms developed to optimize the deployment of WSN, routing protocols, smart power plans and so on. Those algorithms are generally evaluated through simulations on simulators such as NS2, TOSSIM, EmStar or OMNeT++. In [1], the authors reviewed a variety of realistic propagation models for WSNs and discussed the modeling of vegetation propagation model in OMNeT++ simulation platform. They proved that propagation model has strong impact on network performance. However, the aforementioned simulators employ propagation models as simple as free-space model and log-normal model, which are too optimistic and environment independent, and the evaluations of those protocols and algorithms are not rigorous which will lead to inappropriate evaluation and misleading the deployment solution. Therefore accurate RF propagation simulation plays an important role for pre-deployment phase and for the performance evaluation of WSNs.

There are many RF propagation simulation works focused on mobile communication. Measurements have been made in many cities such as Munich scenario [2] and Ilmenau scenario [3]. Furthermore, attenuation parameters are also tested when mobile signal penetrates through obstacles that are made of different materials, such as different thickness of bricks, concrete, metal, etc. However, there are only a few research works focused on modeling RF propagation for ZigBee WSNs, and even fewer works on measuring ZigBee radio propagation in reality. For this reason, recently more researchers focus on modeling radio propagation for WSN and several experiments are made to characterize ZigBee propagation features in both indoor and outdoor environments [4], [5].

In [6], the authors discussed advanced mobile signal tools that might be suitable for estimating 2.4 GHz ZigBee protocol. In order to be accurate and efficient, the tools should have the features like 3D calculation, accurate environment modeling and radio modeling. In [7], the authors characterized wireless channels for indoor propagation at 2.4 GHz, but only direct path is considered. They conclude that free space propagation is unreliable, log-normal model is not completely matched with the trend of curvature of real measurements and the multiwall-floor model is the most reliable and accurate among the discussed models. However, only considering the direct path is unilateral. Diffractions and reflections are also vital in indoor scenarios. A 2.5D outdoor ray launching tool is presented in [8]. The tool is very fast but the resolution is low (7 m) and the computation load is decreased by reducing 3D rays to 3D Line of sight (LOS), 2D Horizontal diffraction and reflection (HDV) and modified 2D Vertical diffraction (VD), therefore the accuracy of simulation result is constraint by the reduced resolution and the types of rays. A 3D ray-optical approach is presented in [9]. The calculation is in 3D, however in order to accelerate the calculation, the method preprocesses the environment by dividing the obstacles into tiles and the edges into segments, and ray paths are limited to only search over the tiles and segments.

As the time consumptions of existing ray tracing methods are decreased by either reducing dimension of rays or reducing

the details of the environment, thus the simulations are efficient while inaccurate. Moreover, modeling of environment is usually time consuming while none of the above mentioned works introduces method to model the environment efficiently and accurately. Therefore, in order to facilitate the simulation, efficient 3D environment modeling, 3D computation and accurate radio modeling is highly demanded.

In this paper, an efficient approach for preprocessing the indoor environment is introduced, the environment is constructed automatically based on image classification and walls are recognized without supervision. The environment database is built by vectorizing the recognized walls and pre-stored attenuation parameters for different materials are automatically assigned to the database once the recognition result is obtained. Besides, a full 3D ray tracing engine together with RF propagation model is proposed, which is sensitive to obstacles including their locations, materials and orders along the ray path. The proposed method does not limit the details of environment or reduce the dimension of rays. Advanced kd-tree space division algorithm and modified polar sweep algorithm are employed to accelerate the tracing of ray paths. Moreover, three different deployments of WSN are carefully planned and measured in the indoor office environment. The simulation results are validated by comparing with measurements, free-space propagation model and log-normal model.

This paper is organized as follows: First, automatic environment reconstruction and pre-processing of database based on image processing algorithms are introduced and reconstructed result is demonstrated (section II), 3D ray-tracing engine and radio propagation model are presented in section III, then measurements are made, results are compared and analyzed in section IV. At the end, conclusions are drawn and future works are described in section V.

II. Automatic Environment Reconstruction and Demonstration

The obstacles in target area cause attenuation along ray path, and result in reflections and diffractions. As discussed in [6], the development trend of advanced simulation tools should contain features like 3D environment modeling and 3D calculation to obtain high accuracy and user-friendly visualization. The automatic 3D outdoor environment reconstruction method proposed in [10] is efficient in modeling the outdoor scenes. The algorithm first recognizes multiple objects from the picture taken from target area, then automatically segments objects and assigns 3D properties including materials, locations and sizes. This method significantly reduces time and labor spent on building 3D environment database. However, the majority of outdoor obstacles are buildings, cars and trees which locate separately and with obvious differences in textures and colors, whereas indoor scenario is much more chaotic: the furniture is sometimes adjacent with walls and are with varieties of textures and shapes; walls and ceiling/floor are conjunct with each other which are difficult to distinguish with high accuracy through similar image classification method. Although there are research works focused on 3D reconstruction in indoor environment, the assumptions on environment are too strong,

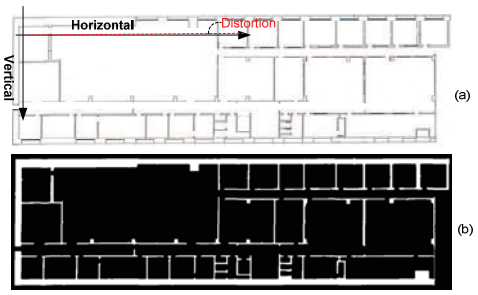


Figure 1 Image calibration and segmentation. (a) scanned map, (b) calibrated and segmented result

such as it should be ”Manhattan world” [11] or floor and ceiling must appear in the picture, therefore their results are not accurate and the methods are not generic enough to be implemented for radio simulation applications. Inspired by the work proposed in [10], a method is developed to automatically model the 3D indoor environment from a scanned 2D map without considering furniture for preliminary results. The method consists of three steps: image calibration and classification step which recognizes the objects from the scanned map; thinning and feature points extraction is to compress the edge information; edge smoothing and vectorizing to build the 3D database.

A. Image calibration and classification

Walls in the maps are normally labeled with black lines as shown in Fig. 1(a). Once the map is scanned, the image is converted from RGB color to gray color. Because floor plan on paper has some distortion and transformation during scanning, the image should be calibrated with 2D horizontal-vertical direction to make sure the axis of map matches well with real world axis. Thereafter, each pixel in the image is checked and pixels that belong to walls are recognized by using a single classifier $h(i)$, where $p(i)$ is the color of i^{th} pixel in the image. If the value is less than v, pixel i is marked by 1 and indicates the wall, otherwise it is marked with 0 which is the background. Hence, a segmentation result is obtained after this step, see Fig. 1(b).

$$h(i) = \begin{cases} 1, & p(i) < v \\ 0, & p(i) \geq v \end{cases} \quad (1)$$

B. Thinning and feature points extraction

In 2D vision of indoor scene, a wall can be expressed by an edge $E < V1; V2 >$ with $V1$ and $V2$ as two endpoints. However, the extracted pixels from the previous step contain redundant information for constructing walls. Thus they are thinned to lines with 1-pixel width based on conventional thinning method and deliver the skeleton of wall efficiently. Afterwards, Harris-corner algorithm [12] is employed to search critical points throughout the segmented pixels and those non-critical points should be eliminated at the end to reduce the redundancy.

C. Smoothing and vectorizing

Critical points are clustered and map is vectorized in this step. Four windows with different directions are used to cluster feature points that belong to the same line. Fig. 2 shows the

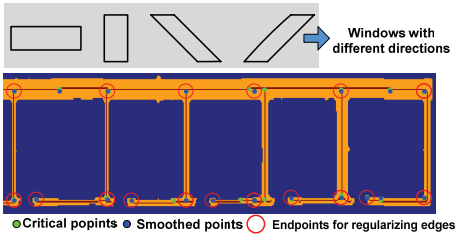


Figure 2 Smoothing and regularization

Figure 3 Vectorization result

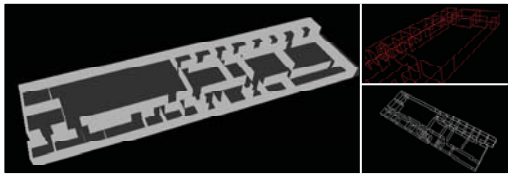


Figure 4 Reconstructed 3D map in different views

shapes of windows and an example of smoothing and regularization. As can be seen, the green dots are critical points obtained from previous step. After applying the windows, the locations of points are adjusted and marked by blue dots. Based on the smoothing algorithm, two points belong to the same cluster and with largest distance are selected as endpoints of an edge. This step reduces the number of unnecessary planes and the information is therefore compressed. The vectorization results are stored in '.txt' file, and the format is shown at the right side of Fig. 3. Vertices in 3D format are expressed as $Vi = [xi;\ heighti;\ yi]$. In this work, all the walls are assumed to be quadrangle and have 4 vertices. However, all the edges are discovered in 2D vision and only $V1 = [x1;\ 0;\ y1]$ and $V2 = [x2;\ 0;\ y2]$ are known. Therefore, once the height of wall is known, $V3 = [x1;\ height;\ y1]$ and $V4 = [x2;\ height;\ y2]$ are generated by assigning value to height. The structure of wall can be expressed by

$$wall = \begin{bmatrix} V_1 & V_2 & V_3 & V_4 \\ material & - & - & - \end{bmatrix}'$$

where $material$ is the index of material such as glass, wood, brick, concrete and so on, it can be assigned by user.

Fig. 3 compares the computed edges with the map. Only two edges have relative big errors (0.35 m) and one wall is missed, the rest edges match well with the map with an average error of 0.13 m. In this example, 86 edges are extracted. As the heights of walls are identically 3 m, they are included automatically to the result. Besides, after the endpoints are fetched, the horizontal range and vertical range of horizontal plane are also obtained, therefore floor and ceiling are added automatically to the result and 88 planes are constructed at the end. The 3D views of the reconstructed indoor environment are demonstrated in Fig. 4. The size of the target space $is\ 57.7\ m \times 3\ m \times 16.7\ m$. The time for constructing this indoor map is less than 1 minute, compared with conventional method which took almost 1 day per person by manually typing 89×4 endpoints, this work significantly reduces the time and labor work on modeling the environment.

III. 3D Ray Tracing Engine and Radio Propagation Modelling

In this section, a 3D ray tracing engine is developed to efficiently search rays and the modeling of radio propagation is introduced. Before tracing rays, the 3D reconstructed scene is loaded to the engine, and the scene is split by using kd-tree algorithm. The target space is divided into small cubes with different volumes to balance the number of obstacles among the cubes. Kd-tree algorithm plays a very important role in accelerating the speed of ray tracing which avoid exhaustive search of ray paths over the entire space.

At the beginning, the environment is polar swept. Conventional polar sweep method is modified by bending the direction of the line whenever intersection occurs. A 3D line is rotated clock wise and bottom up around the transmitter (TX) to discover for each direction the first intersected point and its plane. The rule of reflection is then applied to the line, which bends original direction of the line. Intersected planes are recorded whenever the direction changes at halting points. This procedure is repeated for each candidate direction, and each 3D line terminates shooting after a maximum number of intersections is reached or when the boundary is touched. Therefore, by sweeping the entire scene, all the possible orders of reflection planes and diffraction cones are discovered. The order of reflected plane is stored as a matrix with dimension of $N * depth$, where N is number of possible reflection paths, $depth$ is the maximum depth of reflections predefined in the ray tracing engine. For instance, the $depth = 4$ in Fig. 5, and the reflected plane is expressed as Ref_plane. When the visible plane order exists and the length is less than $depth$, the ID of wall is assigned, then $NULL$ is assigned to the remaining elements:

$$Ref_{plane} = \begin{bmatrix} w7 & NULL & NULL & NULL \\ w1 & w2 & w5 & w6 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

Diffraction happens at the convex edges, the diffraction points are extracted based on the resolution of z direction and $Dif\_cone = \{point;\ visible\_plane;\ invisible\_plane\}$ is a structure used to store all the information of a diffraction cone, where $Dif\_cone.point$ is the location of the diffraction point, $Dif\_cone.visible\_plane$ is the visible plane it belongs to, and $Dif\_cone.invisible\_plane$ is the invisible plane connected with the visible plane, if there is no such a invisible plane, $Dif\_cone.invisible\_plane = NULL$. After the polar sweeping, the rays are traced for each RX, direct path, reflection paths and diffraction paths are selected and the received power strengths (RSS) are calculated separately according to models given in the following subsections.

A. Direct path

A direct ray is launched from TX to RX, and each intersection is recorded when penetrating through the objects.

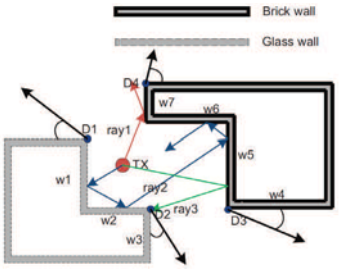


Figure 5 Polar sweep

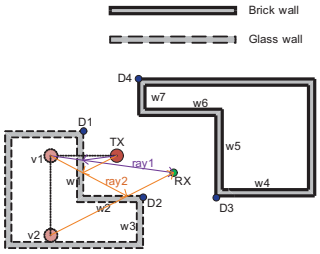


Figure 6 Reflection path searching

The radio propagation model which is developed by us in [10], is used to estimate power loss of direct path between TX and RX.

$$L_p = 10n\,log_{10}d + L_{obstacle} \quad (2)$$

where n is path loss coefficient ranging from 2 to 5. The value of n is decided by the environment, i.e. in free space $n = 2$, in others such as urban or rural environments $2 < n < 5$. For a direct path, d is the 3D distance between TX and RX. $L_{obstacle}$ is the power loss due to obstacles encountered by signal path:

$$L_{obstacle} = \sum_{i=1}^{N} l(i)\alpha(i)^{i-1} \quad (3)$$

it is obtained by accumulating power reduction of each obstacle along the signal path. $l(i)$ is attenuation parameter of the i^{th} obstacle, $\alpha(i)$ ranges from 0 to 1, which is penetration rate of the material of i^{th} obstacle. $\alpha(i)^{(i-1)}$ decreases when i increases, meaning that the first object, with which the signal intersects, produces the most significant power loss.

In the ray tracing engine, instead of calculating the path loss, the RSS of direct path loss RSS_{dir} is computed. It is expressed in watts rather than in dB, resulting

$$RSS_{dir}(d) = \frac{RSS_0 \prod_{i=1}^{N} L_{obstacle(watts)}(i)}{d^n} e^{-i2\pi kd} \quad (4)$$

where RSS_0 is the RSS_{dir} at reference distance. $L_{obstacle(watts)}(i)$ is $L_{obstacle}(i)$expressed in $watts$, k is the wave length of radio, and $k \propto 1/f$, f is the frequency in Hz.

B. Reflection path

Starting from RX, each possible path discovered in Ref_plane is traversed. The procedure is shown in Fig. 6, only two rays are shown as example. According to the second possible path in Fig. 5, $Ref\_plane(2) = [w1\ w2\ w5\ w6]$. For $w1$, virtual source $v1$ is generated by mirroring TX along $w1$, $v2$ is obtained by mirroring $v1$ along $w2$. There are two reflection paths in this plane order, one is through $TX \rightarrow w1 \rightarrow RX$, and the other one is $TX \rightarrow w1 \rightarrow w2 \rightarrow RX$. $w5$ and $w6$ are ignored in current order as there does not exist any ray through $w1$, $w2$ to $w5$ and $w6$. With this searching method, all candidate paths from TX to RX are recorded at the end, and RSSs due to reflections are computed according to the model in Uniform Theory of Diffraction (UTD) [13]. For a RX at point P, its RSS_{ref} reflected from point Q is expressed by:

$$RSS_{ref}(d_{QP}) = RSS_i(Q)Re^{-j2\pi kd_{QP}} \quad (5)$$

where $RSS_i(Q)$ is the RSS of ray incident at Q. R is the reflection coefficient related to the material of encountered plane.

C. Diffraction path

Diffractions happen at the diffraction points found in Dif_cone. When RX is shadowed by planes, diffraction will play a relative important role in RSS. The RSS due to diffraction is calculated by (6),

$$RSS_{dif}(d_{QP}) = RSS_i(Q)De^{-j2\pi kd_{QP}} \quad (6)$$

where Q represents D_2 and D is diffraction coefficient,

$$D = \frac{(1+\cos\theta)}{2\sqrt{2\pi f}}$$

$\theta = |\angle(TX, RX) - \angle(TX, Q)|$ is the angle differential between the ray from TX to Q and the ray from TX to RX. At the end, according to the UTD theory, the RSS at RX is the combination of RSS_{dir}, RSS_{ref} and RSS_{dif} .

IV. Measurements and Result Analysis

Real measurements are realized to verify the simulation method. Cookies sensor nodes[14] developed by CEI are used in the deployments. Cookies are designed as modular architecture of four layers. The communication layer includes a ZigBee module ETRX2 from Telegesis, which is a low power 2.4 GHz band transceiver based on the Ember EM250 SoC ZigBee/IEEE802.15.4 solution. The antenna pattern is near Omni-directional with linear polarization.

In experiment, the transmitter is connected and power supplied by laptop through USB cable, the transmit power is set to be 3 dBm, the sensitivity of antenna is -97 dBm, the receiver is powered by two AA batteries. The transmitter sends packets to the receiver to request link information. The receiver fetches RSSI value of the received packet and replies to the transmitter. By placing the receivers at different locations, different RSSIs are obtained.

Figure 7 Four layers architecture and physical view of the node

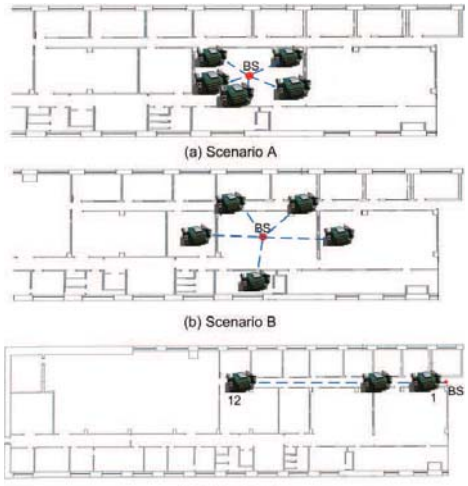


Figure 8 Scenarios


Figure 9 Polar sweep result of TX2 for Scenario A and B

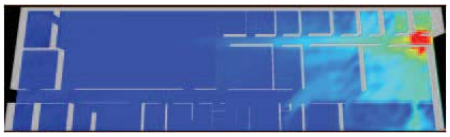


Figure 10 Simulation result of TX1

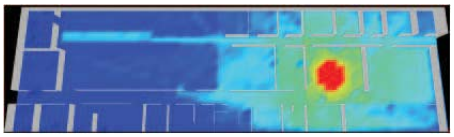


Figure 11 Simulation result of TX2

Three different scenarios are measured as shown in Fig. 8. The Scenario A is realized in a room equipped with several desks and computers. The transmitter is fixed at 1.04 m high above the ground. The receiver is kept at the same height as the transmitter, and RSSIs are measured at five locations as indicated in Fig. 8(a) to verify the propagation model with trivial obstacles, exam the value of path loss coefficient n and $RSS(d_0)$. In scenario B (Fig. 8(b)), the position of transmitter is unchanged, while the receivers are placed further away from transmitter, and are in different rooms. This is to verify the attenuation model of non-trivial obstacles, and obtain empirical penetration parameters. Scenario C (Fig. 8(c)) is realized in the corridor where the communication between TX and RX are LOS, and because the width is much smaller compared with the length, the tunneling effect occurs. The antennas are pointed toward directions with maximum RSSI to minimize the radiation pattern degradation.

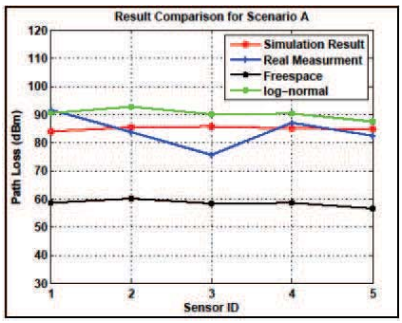
Fig. 9 shows an example of the visible planes to the TX and the diffraction points discovered at the polar sweep step. Actually, the floor and ceiling are also visible to TX, but they are not highlighting for the reason of friendly visualization. As can be seen, there are more than ten rays that could arrive at a RX: directed, reflected, diffracted rays, which are considered without simplification. Fig.10 and Fig. 11 show the simulated RSS maps for those scenarios. When signal path encounters with walls, the RSS is affected significantly. During simulation, some parameters are set to constant, $Lobstacle(watts) = 0.7$, $\alpha = 0.91$, which are tuned by launching simple measurements. $depth = 3$ to maintain the accuracy and reduce the computation consumption as well.

Simulation results are compared with real measurement, free-space model and log-normal model. The resolution is set to be $64 \times 20 \times 2$, hence there are 2560 candidate RXs to be calculated. The ray tracing engine runs in a PC with Intel Core i5 CPU, 2.8 GHz. The time for constructing kd-tree space division is 46 ms, for polar sweeping is 36 s, and for ray tracing is 74 s, which is around 29 ms per candidate RX. Conventional 2D ray tracing methods take average 19 ms which is less but comparative with 3D. However, it should be noticed that there are far more rays in 3D than in 2D, and that is the main reason why many researchers only discuss 2D and 2.5D ray tracing methods rather than 3D ray tracing.

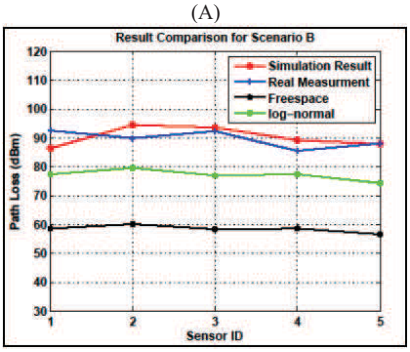
Scenario A is a typical indoor environment without important obstacles between transmitter and receiver. Through the measurement, parameters are tuned: $n = 3$, $RSS(d_0) = 10^{-6}$, which are then used for all the three scenarios. Fig. 12(A) compares ray tracing method with free-space path loss model, log-normal model and real measurement. Free-space model is too optimistic in estimating indoor propagation with mean error $mean_{error} = 27.87$, standard deviation $std_{error} = 4.72$. While log-normal is much more suitable than free-space model and $mean_{error} = 4.26$, standard deviation $std_{error} = 2.29$. The ray tracing method has $mean_{error} = 4.76$, and $std_{error} = 3.76$ which is slightly worse than the log-normal model. As there is no important obstacle and the dominant contributions of RSS come from direct paths and reflection paths. However, because of the Gaussian random variable in the log-normal model, its performance is not always better than ray tracing model at each simulation time.

In Scenario B, RX are in different rooms than TX, all possible paths may make contributions to RSS. In this scenario, the effects of reflections and diffractions are more obvious than in scene A, see Fig. 12(B). In this case, ray tracing has $mean_{error} = 3.22$, and $std_{error} = 2.40$; free-space model has $mean_{error} = 31.28$, and $std_{error} = 2.93$; Log-normal has $mean_{error} = 12.50$, and $std_{error} = 3.13$. As log-normal model does not consider obstacles between TX and RX, the result is much worse than the ray-tracing model.

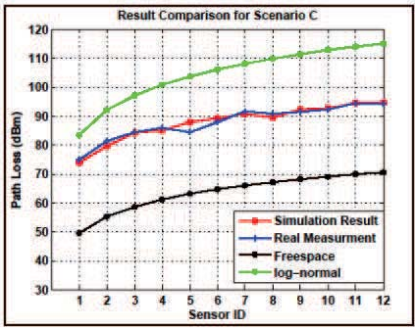
In Scenario C, there is no obstacle between TX and RX. However, due to the narrow width of corridor, reflections and diffractions play more important role than that in A and B. The ray tracing result fits best with the real measurement even in important high level fluctuations. Although $n = 3$ in both raytracing and log-normal model, the slopes of curves are quite different between them while the slope of free-space model is almost the same as that of ray-tracing model. In this



(A)



(B)



(C)

Figure 12 Results and comparisons tested scenarios

log-normal model has $mean_{error} = 16.73$, and $std_{error} = 4.14$; free-space model has $mean_{error} = 24.22$, and $std_{error} = 1.41$. Through the comparisons, ray-tracing method is proved to be robust for different indoor scenarios.

V. Conclusions and Future Works

A 3D radio propagation simulation method for indoor ZigBee WSNs has been presented in this paper. As discussed, there are mainly three contributions in this work:

First, it is the first time that the automatic indoor reconstruction is employed for simulating RF propagation, the result is accurate and efficient. Secondly, ray tracing is developed in full 3D based on kd-tree space division algorithm and modified polar sweep algorithm, and the speed is comparative with conventional 2D ray tracing methods. Besides, the propagation model emphasizes not only the materials of obstacles but also their locations. Hence, the performance of simulation is robust and accurate compared with conventional propagation models. Last but not the least, the experiment data are available in this paper, which can be used for comparisons by other researchers. The time consumptions are recorded for all the steps of ray tracing, the measurements on RSSs for different deployments and the attenuation parameters of different materials are also available in this paper.

In the future, obstacles such as closets and desks will be automatically modeled in order to be more accurate and efficient in environment modeling. Besides, the method will be tested for more indoor maps and more measurements on ZigBee propagations will be realized.

Acknowledgment

The authors would like to acknowledge the support of ARTEMIS JU and Spanish Ministry of Industry and commerce for WSN DPCM project under grant ART-010000-2011-1.

References

[1] L. M. Kamarudin, R. B. Ahmad, B. L. Ong, F. Malek, A. Zakaria, and M. A. M. Arif, “Review and modeling of vegetation propagation model for wireless sensor networks using omnet++,” in Proc. Second Int Network Applications Protocols and Services Conf, 2010, pp. 78–83.

[2] COST231, “Evolution of digital mobile radio: Cost 231 view on the evolution towards 3rd generation systems,” Commission of the European Communities and COST Telecommunications, Brussels, Tech. Rep., 1999.

[3] J. Kenyeres, S. Sajban, P. Farkas, and M. Rakus, “Indoor experiment with wsn application,” in Proc. 33rd Int MIPRO Convention, 2010, pp. 863–866.

[4] R. M. Pellegrini, S. Persia, D. Volponi, and G. Marcone, “Rf propagation analysis for zigbee sensor network using rssi measurements,” in Proc. 2nd Int Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE) Conf, 2011, pp. 1–5.

[5] V. Kolar, S. Razak, P. Mahonen, and N. B. Abu-Ghazaleh, “Measurement and analysis of link quality in wireless networks: An application perspective,” in Proc. INFOCOM IEEE Conf. Computer Communications Workshops, 2010, pp. 1–6.

[6] M. Korkalainen and M. Sallinen, “A survey of rf-propagation simulation tools for wireless sensor networks,” in Proc. Fourth Int Sensor Technologies and Applications (SENSORCOMM) Conf, 2010, pp. 342–347.

[7] T. Chrysikos, G. Georgopoulos, and S. Kotsopoulos, “Wireless channel characterization for a home indoor propagation topology at 2.4 ghz,” in Proc. Wireless Telecommunications Symp. (WTS), 2011, pp. 1–10

[8] Z. Lai, H. Song, P. Wang, H. Mu, L. Wu, and J. Zhang, “Implementation and validation of a 2.5d intelligent ray launching algorithm for large urban scenarios,” in Proc. 6th European Conf. Antennas and Propagation (EUCAP), 2012, pp. 2396–2400.

[9] T. Rautiainen, R. Hoppe, and G. Wolfle, “Measurements and 3d ray tracing propagation predictions of channel characteristics in indoor environments,” in Proc. IEEE 18th Int. Symp. Personal, Indoor and Mobile Radio Communications PIMRC 2007, 2007, pp. 1–5.

[10] D. He, G. Liang, J. Portilla, and T. Riesgo, “A novel method for radio propagation simulation based on automatic 3d environment reconstruction,” in Proc. 6th European Conf. Antennas and Propagation (EUCAP), 2012, pp. 445–1449.

[11] J. M. Coughlan and A. L. Yuille, “Manhattan world: compass direction from a single image by bayesian inference,” in Proc. Seventh IEEE Int Computer Vision Conf. The, vol. 2, 1999, pp. 941–947.

[12] C. Harris and M. J. Stephens, “A combined corner and edge detector,” in Proc. Alvey Vision Conference, 1988, pp. 147–152.

[13] D. A. McNamara, C. W. I. Pistorius, and J. A. G. Malherbe, Introduction to the Uniform Geometrical Theory of Diffraction. Boston, MA: Artech House, 1990.

[14] J. Portilla, A. D. Castro, E. D. L. Torre, and T. Riesgo, “A modular architecture for nodes in wireless sensor networks,” Journal of Universal Computer Science, vol. 12, no. 3, pp. 328–339, 2006.