Research Article

A Comprehensive Propagation Prediction Model Comprising Microfacet Based Scattering and Probability Based Coverage Optimization Algorithm

A. S. M. Zahid Kausar, Ahmed Wasif Reza, Lau Chun Wo, and Harikrishnan Ramiah

Department of Electrical Engineering, Faculty of Engineering, University of Malaya, 50603 Kuala Lumpur, Malaysia

Correspondence should be addressed to Ahmed Wasif Reza; awreza98@yahoo.com

Received 13 February 2014; Revised 13 July 2014; Accepted 13 July 2014; Published 18 August 2014

Academic Editor: Nirupam Chakraborti

Copyright © 2014 A. S. M. Zahid Kausar et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Although ray tracing based propagation prediction models are popular for indoor radio wave propagation characterization, most of them do not provide an integrated approach for achieving the goal of optimum coverage, which is a key part in designing wireless network. In this paper, an accelerated technique of three-dimensional ray tracing is presented, where rough surface scattering is included for making a more accurate ray tracing technique. Here, the rough surface scattering is represented by microfacets, for which it becomes possible to compute the scattering field in all possible directions. New optimization techniques, like dual quadrant skipping (DQS) and closest object finder (COF), are implemented for fast characterization of wireless communications and making the ray tracing technique more efficient. In conjunction with the ray tracing technique, probability based coverage optimization algorithm is accumulated with the ray tracing technique to make a compact solution for indoor propagation prediction. The proposed technique decreases the ray tracing time by omitting the unnecessary objects for ray tracing using the DQS technique and by decreasing the ray-object intersection time using the COF technique. On the other hand, the coverage optimization algorithm is based on probability theory, which finds out the minimum number of transmitters and their corresponding positions in order to achieve optimal indoor wireless coverage. Both of the space and time complexities of the proposed algorithm surpass the existing algorithms. For the verification of the proposed ray tracing technique and coverage algorithm, detailed simulation results for different scattering factors, different antenna types, and different operating frequencies are presented. Furthermore, the proposed technique is verified by the experimental results.

1. Introduction

Nowadays, indoor wireless communication becomes more and more popular in communication field. Because of increasing demand in this field, an effective propagation prediction technique and optimized coverage algorithm are required in order to support the demand by using the minimum number of transmitters (T_x s) and at the same time achieving the maximum indoor wireless coverage. Though there are a number of existing research works based on ray tracing for propagation prediction [1–7], most of them have not mentioned anything about the coverage. Therefore, researchers are still in need of an efficient and integrated method, which can serve for propagation prediction and coverage optimization.

The main problem for the ray tracing based propagation prediction model is the ray-object intersection test. This test consumed the most time and resources in a ray tracing method [8, 9]. Intersection test is performed every time after a new ray is generated for finding whether there is a ray-object intersection or not. Hence, if all objects participate in this test, the ray tracing time consumed will be extremely high. To accelerate the ray tracing technique, various methods, such as angular sectoring [10], KD-tree, octree, quadtree [4, 5, 11], and a preprocessing method [8], are proposed. However, the existing models, such as shooting and bouncing ray (SBR) [4], bidirectional path tracing (BDPT) [12], brick tracing (BT) [13], ray frustums (RF) [14], prior distance measure (PDM) [8], and space division (SD) [15] techniques,

require higher prediction time due to complex algorithms used and limitations of the used techniques. Moreover, the prediction accuracy is not so high. In SBR technique, double ray counting error occurs for the receivers (R_x s) situated in two successive ray cone areas. The BDPT technique shows incorrect results for single floor multiple room environments and takes a lot of time to create the ray paths. The BT technique shows inaccuracy for corner bricks because of the truncation of the slab, which results in erroneous analytic reflection and transmission coefficient. In RF technique, a large computer memory is required for complex environments to store the huge amount of triangular frustums, which results in slow performance. In SD technique, all the IDs of the unique cells are stored in a single list and the full list has to search for finding a specific cell during simulation. This will consume a lot of time and increase the execution time. In PDM method, a preprocessing operation is needed for the environment. This type of preprocessing makes the overall process more complex. None of the above techniques are accompanied by the coverage optimization technique.

Considering all of the drawbacks of the existing techniques, this paper introduces a new method by including the rough surface scattering. This proposed ray tracing method is based on Adelson-Velskii and Landis (AVL) tree data structure, dual quadrant skipping (DQS) technique, and closest object finder (COF). The AVL tree has a lower data searching time and it is used for efficiently handling different information relative to the objects and environment. Both of the DQS and COF techniques are newly introduced and described here. Both of the techniques help to reduce the ray tracing time by eliminating the unnecessary objects and enhancing the ray-object intersection test. Furthermore, our proposed microfacet based scattering method is not dependent on specific environment and is able to figure out the scattering field in all likely directions. This makes the proposed ray tracing method more accurate.

Along with the ray tracing technique, a new coverage optimization algorithm is also introduced here where the probability is used to find out the most suitable T_x to be selected in order to achieve the optimum solution for indoor wireless coverage area. The probability of each T_x is directly affected by the T_x sampling pattern. The proposed algorithm introduces two types of probability that need to be taken into consideration, that is, intraprobability and interprobability. The concepts of probability will be explained in more detail in the next section. In order to support the probability concept, multilevel technique has been applied in the proposed algorithm where each T_x sampling pattern is viewed level-by-level instead of T_x -by- T_x . By applying multilevel technique in the proposed algorithm, it provided faster computation time; this is due to less T_x taking part in probability calculation for certain number of sampling points. For achieving better performance, genetic algorithm (GA) and depth first search (DFS) are used along with the probability theory for finding the optimum wireless coverage. GA is a widely used [16–19] algorithm for optimization of different electromagnetic problems. Here, it is used to optimize the number of T_x s needed for covering the whole area. It will also optimize the position of the necessary T_x s.

In DFS, the node which is generated from the E -node is called a live node. E -node refers to the node where children are currently generated. The E -node is selected from various live nodes in the same level based on the probability. T_x having the highest probability will be selected as E -node and during each DFS process, every T_x can be selected as E -node at least once. To minimize the required computation time (time complexity) of the proposed coverage algorithm, changes have been made for the existing bounding and termination concepts that were proposed by the existing algorithm [19]. Basically, the bounding function uses for updating the latest probability of each new subbranch, hence improving the accuracy to determine the number of T_x s required at corresponding positions. Besides, a termination criterion is used to avoid repeating select nodes, which had been selected as E -node before. The proposed algorithm generates less number of nodes in the search tree and further reduces the computation time by using this bounding function and termination criteria. Some analyses and comparisons have been made and the results prove that the proposed algorithm is more efficient than the existing algorithms [19, 20] in terms of space tree generated and also the time complexity.

2. Proposed Ray Tracing Technique

2.1. Technique for Achieving Balanced AVL Tree. For data storing and retrieving purpose, a dynamically height balanced binary search tree, named AVL tree, is used. An AVL tree has two basic properties: the height of the subtrees of every node differs by at most one and each subtree is an AVL tree. An AVL tree maintains a $O(\log n)$ search time, while the addition and deletion operation also take $O(\log n)$ time (where n is the number of objects). This timing is almost similar with another self-balancing tree, namely, red-black tree. However, the difference between them is the limiting height. For a tree of n objects, the maximum height of an AVL tree is strictly less than [21]

$$\begin{aligned} \log_{\varphi}(\sqrt{5}(n+2)) - 2 &= \frac{\log_2(\sqrt{5}(n+2))}{\log_{\varphi} 2} - 2 \\ &= \log_{\varphi}(2) \log_2(\sqrt{5}(n+2)) - 2 \\ &\approx 1.44 \log_2(n+2) - 0.328, \end{aligned} \quad (1)$$

where φ is the golden ratio.

At the same time, the maximum height of a red-black tree is [22]

$$h = 2 \log_2(n+1). \quad (2)$$

Therefore, we can say that the AVL trees are more rigidly balanced than red-black trees. For this reason, the AVL tree data structure is chosen.

The data insertion technique in an AVL tree is identical to a binary search tree, where it is done by expanding a peripheral node. For maintaining balance, information of the balance factor will have to be stored in every node. This balance factor will maintain the balance efficiently after each

FIGURE 1: Sample environment for showing the AVL tree creation.

FIGURE 2: Balancing of AVL tree by a single rotation: (a) insertion of new node, (b) non-AVL tree, and (c) balanced AVL tree after single rotation.

insertion or deletion operation. The balance factor, bf, can be represented as

$$\text{bf} = \text{height of left subtree } (hL_{\text{subtree}}) - \text{height of right subtree } (hR_{\text{subtree}}). \quad (3)$$

This bf indicates if the two subtrees are in the same height or not. The bf of a height balanced binary tree can be one of the values $-1, 0, +1$. An AVL node is called left heavy when $\text{bf} = -1$, equal heavy when $\text{bf} = 0$, and right heavy when $\text{bf} = +1$. After insertion of each new node, the balance factors have to be updated. If the balance factor becomes less than -1 or greater than $+1$, the tree becomes unbalanced and rotation process will be undertaken for making balanced AVL tree.

Here, Figure 1 represents a sample environment of 10 objects. The creation of AVL tree from this sample environment is presented here in Figures 2 and 3. Figure 2 shows a single rotation for making balance after insertion of 6th object and Figure 3 shows balancing with double rotation after insertion of the 10th object in the sample environment of Figure 1. The further insertion of objects can be done by following this technique.

2.2. Microfacet Based Scattering Model. In this section, the computational model for computing the scattering field of

rough surface has been presented. This model is based on Kirchhoff approximation (KA) [18, 23], which is a well-known method. Hence, the rough surface is decomposed into small planes that are nearly tangent to the roughness and these are called microfacets. Figure 4 represents the profile of a rough surface, a random tangent plane on it, and the notations, which are going to be used for incident and scattered fields.

An incident plane wave irradiates the decomposed rough profile. Thus, the same amount of incident plane wave is received by each microfacet and the surface reflects it towards its own specular direction. As well as the smooth surface, this specific specular direction can be defined by incident angle θ_i and alignment of the indigenous normal η of a microfacet.

Therefore, the scattering field can be computed as

$$\vec{F}_{//,\perp}^s = R_{//,\perp}(\theta_i, \theta_s) \vec{F}_{//,\perp}^i e^{-j\varphi} e^{-j(\phi_i + \phi_s)}, \quad (4)$$

where $\vec{F}_{//,\perp}^s$ is the scattered field and $\vec{F}_{//,\perp}^i$ is the incident field in both polarizations, $R_{//,\perp}(\theta_i, \theta_s)$ is the Fresnel reflection coefficient between incident θ_i and scattered θ_s directions, and $e^{-j(\phi_i + \phi_s)}$ is the shifting of phase caused by the free space propagation distance after ϕ_s and before ϕ_i . The term $e^{-j\varphi}$ represents the shifting of phase because of the height h of the microfacet regarding the global mean value, which is usually set at $h = 0$ [12]. It can be written as

$$\varphi = \frac{4\pi}{\lambda} h \cos(\theta_i). \quad (5)$$

Now, a rational sum of all the aid of the scattered fields summarized in a minute solid angle $d\theta$ around a definite direction θ_s will have to be formulated. In reality, the orientation of many microfacets can be the same but, depending on the global mean value, their heights are not automatically same. If n numbers of microfacets among N numbers are well directed, the rational sum is equivalent to a vector sum for each component $(\vec{x}, \vec{y}, \vec{z})$ of the scattered fields [12]:

$$\vec{F}_{//,\perp}^s = \sum_{i=1}^n \vec{F}_{//,\perp,i}^s, \quad (6)$$

where $\vec{F}_{//,\perp,i}^s$ is the scattering field for both of the polarizations of the i th microfacet.

As a result of the plane wave propagation condition, the simplified scattering field for the i th microfacet can be written as

$$\vec{F}_{//,\perp,i}^s = R_{//,\perp,i}(\theta_i, \theta_s) e^{-j\varphi_i} \vec{F}_{//,\perp,i}^i e^{-j(\phi_i + \phi_s)} = C^{te} e^{-j(\phi_i + \phi_s)}. \quad (7)$$

Now, the whole scattering field around a particular direction will be expressed for the parallel element. The similar interpretation can be used for the perpendicular polarization. So, the scattering field of the i th microfacet in parallel polarization can be expressed as

$$\vec{F}_{//,i}^s(\theta_s) = \begin{cases} R_{//}(\theta_i, \theta_s) \frac{1}{N} \cos(\theta_s) e^{-j\varphi_i} C^{te} & \vec{x} \\ 0 & \vec{y} \\ R_{//}(\theta_i, \theta_s) \frac{1}{N} \sin(\theta_s) e^{-j\varphi_i} C^{te} & \vec{z}. \end{cases} \quad (8)$$

FIGURE 3: Balancing of AVL tree by double rotation: (a) insertion of new node, (b) non-AVL tree, (c) 1st rotation, and (d) 2nd rotation and balanced AVL tree.

FIGURE 4: Notations used in the KA based scattering model.

Then, the vector sum in $d\theta$ around $d\theta_s$ direction for n contributions becomes

$$\vec{F}_{//,i}^s(\theta_s) = \begin{cases} C^{te} R_{//}(\theta_i, \theta_s) \frac{1}{N} \cos(\theta_s) \sum_{i=1}^n \{e^{-j\varphi_i}\} & \vec{x} \\ 0 & \vec{y} \\ C^{te} R_{//}(\theta_i, \theta_s) \frac{1}{N} \sin(\theta_s) \sum_{i=1}^n \{e^{-j\varphi_i}\} & \vec{z}. \end{cases} \quad (9)$$

If we set up the ratio n/N in (9), the scattering field will become

$$|\vec{F}_{//}^s(\theta_s)| = R_{//}(\theta_i, \theta_s) C^{te} \frac{n}{N} \frac{1}{n} \sum_{i=1}^n e^{-j\varphi_i}, \quad (10)$$

where the ratio n/N signifies the probability of having n number of microfacets from N possible numbers in θ_s direction and $(1/n) \sum_{i=1}^n e^{-j\varphi_i}$ represents the mean attenuation due to the shifting of phase. Therefore, the scattering field of (10) can be written with replacement of n/N by the probability density function $p(\theta_s)$ as

$$|\vec{F}_{//}^s(\theta_s)| = R_{//}(\theta_i, \theta_s) C^{te} p(\theta_s) \frac{1}{n} \sum_{i=1}^n e^{-j\varphi_i}. \quad (11)$$

The scattering coefficient $\sigma^{//,\perp}(\theta_s)$ can be deduced from (11) for both polarizations, which provides the ratio between

the scattered power in a solid angle $d\theta$ around θ_s and the incident power:

$$\sigma^{//,\perp}(\theta_s) = |\vec{F}_{//,\perp}^s(\theta_s)| |\vec{F}_{//,\perp}^i(\theta_s)|^*. \quad (12)$$

2.3. Proposed Optimization Techniques for Ray Tracing. After creating the data structure tree, it is necessary to find the objects, which are taking part in intersection test. We have sorted the necessary objects in two different techniques. First, we have used the proposed DQS technique to find a group of objects according to the ray direction. Then, the COF will find the nearest object from that particular group of objects and that nearest object will take part in intersection test. These two acceleration techniques will reduce the intersection test time by finding the exact object.

The DQS technique will reduce the prediction time by reducing the number of considered objects due to each intersection test. From Figure 5, we can easily describe the DQS technique. Suppose a ray is generated from the T_x and intersects with an object at the position (X_1, Y_1) . According to the position of the objects in the environment with reference to the point (X_1, Y_1) , the environment will be divided into four quadrants: I, II, III, and IV. The distribution of the objects into different quadrants is illustrated in Figure 5.

Now assume the object is a nontransparent object and no diffraction is occurring. Hence, the ray will show normal reflection or scattered reflection. Therefore, there is no possibility for the reflected ray to go behind the object. That means that, for the next intersection test, we have no need to consider the objects of quadrants I and IV. Thus, all of the objects in the regions I and IV will be skipped for the next intersection test. This saves almost half of the prediction time for the next intersection test. Now assume, after reflection of the ray at (X_1, Y_1) , the ray intersects with another object at the position (X_2, Y_2) . Again, the whole environment will be divided into four quadrants based on (X_2, Y_2) . After reflecting on (X_2, Y_2) , the ray will go to the front of that object. Again, the shaded portion of the back side of that object will be skipped, which means quadrants II and III will be skipped. If there is refraction (for transparent object) or diffraction instead of reflection, the opposite portions of the quadrant (quadrants II and III) will be skipped and thus will reduce the prediction time.

FIGURE 5: (a) Object skipping using DQS technique in a sample environment and (b) object allocation due to DQS.

By applying the DQS technique, a group of possible intersecting objects can be found and, among them, some objects (e.g., 39th, 40th, and 41st) are parallel. All of them will not intersect with the ray, but only the nearest object will take part in the intersection test. To find the nearest object, we have introduced the COF. The COF is composed of two different phases: first, the creation of an effective artificial surface (EAS) inside each of the possible intersecting objects found by DQS and second finding the distances between the ray source and different EAS by using the “Pythagoras Theorem.”

The EAS as shown in Figure 6(a) is defined as the effective surface within an object, which is used for calculating ray-object intersection. In this concept, a new invisible or imaginary surface is created inside the 3D object at a point where the ray contacted with the object. This surface will be a 2D surface and it is created at the middle of the object. It is introduced with the aim of simplifying the algorithm and, hence, reducing the computational complexity. Using the six vertex points of a cuboid, the software developed in this work will calculate the coordinates of the EAS.

Assuming k , l , m , and n are the four vertices of the EAS and C_1 , C_2 , C_3 , C_4 , C_5 , C_6 , C_7 , and C_8 are the eight vertices of the cube or cuboids, then the coordinate of k of the ICS can be found as

$$X_k = \frac{X_{C_1} - X_{C_3}}{2}, \quad (13)$$

$$Y_k = Y_{C_1} = Y_{C_3},$$

and l , m , and n coordinates are determined in the same manner as the k coordinate.

Now, if the coordinate of k is (x_1, y_1) and coordinate of n is (x_2, y_2) , then the coordinate (X_1, Y_1) of the middle point of kn face will be as

$$X_1 = \frac{(x_1 + x_2)}{2}, \quad (14)$$

$$Y_1 = \frac{(y_1 + y_2)}{2}.$$

Now, suppose the middle of the EAS of the 39th, 40th, and 41st objects is A , D , and E , respectively, and (X_1, Y_1) is

FIGURE 6: (a) Distance calculation between ray source and EAS. (b) Intersection point calculation between ray and a surface.

the coordinate of A. These points will be used for calculating the distance between the objects and ray source point (X_2, Y_2) . This will be done by using the “Pythagoras Theorem.”

By extending the points A and B, a right angled triangle ABC will be formed (Figure 6(a)). The coordinates of the point C will be (X_1, Y_2) . Now, by applying the “Pythagoras Theorem” in the right angled triangle, we found

$$AB^2 = AC^2 + BC^2 = (Y_2 - Y_1)^2 + (X_2 - X_1)^2. \quad (15)$$

If the distance between A and B is D_1 , then, from (15), we found

$$D_1 = \sqrt{dX^2 + dY^2}, \quad (16)$$

where dX is the difference between the X-coordinates of A and B and dY is the difference between the Y-coordinates of A and B.

By this way, distance between B and other middle points will be calculated, which can be D_2 and D_3 , respectively, in the present case. Then, after comparing the distances, the nearest object is chosen and, in the above case, 40th object is found as the nearest. As the nearest object, this one will now be used for an intersection test to find the intersection point. After finding the nearest object, the intersection test is done for getting the exact ray-object intersection point. Based on this intersection point, the next ray shooting decision will be taken from this point. The occurrence of reflection or refraction is also dependent on this point.

As we have used 3D cuboids for representing the objects, there is a possibility of intersection between the ray and the six faces of the cuboids. Depending on the position of the object with respect to the ray shooting point, the exact face

is chosen for finding the intersection point. As an example, if an object is in the first quadrant then the ray will possibly hit the left or back surface of that object and in that case, only these surfaces will be used for finding the intersection point, as demonstrated in Figure 6(b). Similarly, right or back surface for the second quadrant, right or front surface for the third quadrant, and left or front surface for the fourth quadrant, respectively, will be used. In case of the 34th object of Figure 5, right or back surface will have to be tested for finding the intersection point and it is shown in Figure 6(b). According to vector notation, the plane π is expressed as the set of points P for which

$$(P - P_0) \cdot n = 0, \quad (17)$$

where n is a normal vector to the plane and P_0 is a point on the plane.

According to vector algebra, the line of the ray can be expressed as

$$P = dI + I_0, \quad d = R, \quad (18)$$

where I is a vector in the direction of the line, I_0 is a point on the line, and d is a scalar in the real number domain.

Now, by substituting (18) into (17), we get

$$(dI + I_0 - P_0) \cdot n = 0. \quad (19)$$

By distributing n , (19) becomes

$$dI \cdot n + (I_0 - P_0) \cdot n = 0. \quad (20)$$

Now, by solving (20), we get the value of d :

$$d = \frac{(P_0 - I_0) \cdot n}{I \cdot n}. \quad (21)$$

No intersection will exist if the line is parallel to the plane and starts outside the plane. In this case, (21) denominator will be zero and the numerator will be nonzero. The line will intersect everywhere with the plane if the line is parallel to the plane and starts inside the plane. In this case, both the numerator and denominator of (21) will be zero. In all other cases, the line intersects the plane once and d represents the intersection as the distance along the line from I_0 ; that is, $dI + I_0$. Now, this intersection point ($P_{\text{int}}(X, Y, Z)$) will act as the source for that particular ray and reflection, refraction, diffraction, or scattering will occur at that point according to the object property and the ray will proceed to a particular direction. Based on that direction, again, DQS will apply at the intersection point and the next object will be found for a ray-object intersection by applying the COF technique. This process will continue and, at the end, the ray will be counted either as a valid signal received by R_x or an invalid signal.

2.4. Formulation of Time Complexity and Received Power. As we have mentioned earlier, for M number of objects, the search operation of an AVL tree can be implemented in $O(\log n)$ time. Based on (1), the time complexity of the proposed technique can be described as below.

Let M be the number of objects, let N be the number of surfaces of each 3D object, and let S be the intersection testing time for the proposed method. If R numbers of intersections are required to predict each significant ray, then the total intersection testing time can be calculated by the following equation:

$$S_{\text{proposed}} = R \times N \times O(\log_2 M). \quad (22)$$

Moreover, according to DQS, the proposed method can omit a significant amount of objects during each intersection test. Let M_{DQS} be the average number of omitted objects due to DQS. Now, (22) becomes

$$S_{\text{proposed}} = R \times N \times O(\log_2 (M - M_{\text{DQS}})). \quad (23)$$

Furthermore, the COF technique also skips objects during intersection test. Suppose M_{COF} is the number of objects skipped by COF technique. Thus, the equation for intersection time will be

$$S_{\text{proposed}} = R \times N \times O(\log_2 (M - M_{\text{DQS}} - M_{\text{COF}})). \quad (24)$$

For similar cases, the intersection testing time by using the existing techniques [12, 24, 25] can be found as

$$S_{\text{existing}} = R \times N \times O(M). \quad (25)$$

Accordingly, the time complexity of SBR technique is [16]

$$S_{\text{SBR}} = R \times N \times O(\log_2 (M - M_{\text{MT}})), \quad (26)$$

where M_{MT} are the skipped objects due to mailbox technique.

Time complexity for BT and BDPT techniques can be found as [16]

$$S_{(\text{BT\&BDPT})} = R \times N \times O(M). \quad (27)$$

The time complexity of RF technique is [16]

$$S_{\text{RF}} = R \times N \times O(\log_2 (M - 2^H)), \quad (28)$$

where 2^H is the order of quadtree.

Time complexity for PDM and SD techniques is [16]

$$S_{\text{PDM}} = R \times N \times O(M - M_{\text{PDM}} - M_{\text{BSM}}), \quad (29)$$

where M_{PDM} and M_{BSM} are the objects skipped due to prior distance measures and bounding spheres method, respectively.

And

$$S_{\text{SD}} = R \times N \times O(M - M_{\text{SD}}), \quad (30)$$

where M_{SD} are the objects skipped by space division.

The received power at an observation point is calculated with Friis transmission formula. For three-dimensional modeling, three-dimensional directivity data of transmitting and receiving antennas are required, which can be interpolated from the measurement data along the E - and H -planes [26].

If an observation point is included in a ray frustum, a ray path is formed, which traces back to the vertex of the frustum. If the ray hits the reflecting triangle or the diffracting edge that generates the frustum, the ray continues to trace back to the apex of the parent ray frustum. The process is recursively carried out until the ray hits the transmitting antenna. If the ray is intercepted by obstacles in the course of the ray tracing, the received power at the point is assumed to be zero. When the back-traced ray hits the source, the directivity n of the transmitting antenna along the direction of the ray is utilized and the radiated electric field is obtained by [14].

Consider

$$E_{\text{inc}} = \sqrt{\frac{\eta P_T}{4\pi} G_T(\theta_T, \phi_T)} \cdot \frac{h_{T,v} \hat{\theta}_T + h_{T,h} \hat{\phi}_T}{R} e^{-jkR}, \quad (31)$$

where P_T is the transmitted power and η is the wave impedance of the free space. The gain along the ray direction (θ_T, ϕ_T) is $G_T(\theta_T, \phi_T)$. Unit vectors $\hat{\theta}_T$ and $\hat{\phi}_T$ represent the vectors along the elevation and azimuth directions, respectively. The values $h_{T,v}$ and $h_{T,h}$ represent the polarization components.

The incident electric field generated from the transmitting antenna undergoes reflection, transmission, and diffraction. The received electric field of a ray that arrives at the receiving antenna after many combinations of reflections, diffractions, and transmissions can be found as

$$E_{\text{received}} = \left[\prod A(S_{i-1}, S_i) (\bar{R}_i \text{ or } \bar{T}_i \text{ or } \bar{D}_i) e^{-jkS_i} \right] \cdot E_{\text{inc}}, \quad (32)$$

where $A(S_{i-1}, S_i)$ is the divergence factor that accounts for the magnitude changes after i th scattering. The dyadic quantities $\bar{R}_i$, $\bar{T}_i$, and $\bar{D}_i$ are reflection, transmission, and diffraction.

The received voltage is the sum of multipath signals that have propagated along actual ray paths. The voltage can be obtained as

$$V_R = \lambda \sqrt{\frac{Z_0}{4\pi}} \sum G_R(\theta_R, \phi_R) \cdot [(h_{R,v} \hat{\theta}_R + h_{R,h} \hat{\phi}_R) \cdot E_{\text{received}}], \quad (33)$$

where Z_0 is the characteristic impedance of a receiver and $G_R(\theta_R, \phi_R)$ is the gain along the ray direction (θ_R, ϕ_R) . Unit vectors $\hat{\theta}_R$ and $\hat{\phi}_R$ represent vectors along the elevation and azimuth directions seen from the receiving antenna coordinate, respectively. The values $h_{R,v}$ and $h_{R,h}$ represent the polarization components. The received power can be obtained as

$$P_R = \frac{|V_R|^2}{Z_0}. \quad (34)$$

3. Proposed Coverage Algorithm

The basic idea of the proposed algorithm is to obtain the minimum number of T_x s required to achieve the optimum wireless coverage. To achieve this goal, the concept of T_x s' probability to be selected for each sampling point has been introduced and incorporated with GA and DFS to optimize the indoor wireless coverage. Here, a sampling point is a point where it collects the signals coming from the surrounding T_x s. Ray tracing is used to generate different coverage patterns.

In this study, the DFS progresses by expanding the first level of sampling point nodes of the tree that appear and finding out the node with a higher probability and searching from level to level until all the sampling point is covered; then, searching process will be stopped. The key word of the proposed algorithm is the probability. For ease of understanding, the used notations are given here.

- (i) t_i is the i th T_x . Assume each of the T_x s only covers one of the sampling points S . Therefore, the number of T_x s required to cover all the sampling points should be equal to or less than the total number of sampling points. Hence, $1 \leq i \leq n$, if the number of sampling points in the indoor area is n .
- (ii) C_i is the coverage pattern of t_i .
- (iii) L_i is the sampling point level for level i th.
- (iv) P_{ij} is the probability of t_i to be selected for L_j .
- (v) T_{P_i} is the total probability to be selected for t_i .
- (vi) R_i is the t_i to be removed.

If C' and C'' are the two coverage patterns, then the resultant pattern C^* will be as

$$C^* = \left\{ C' \cup C'' \mid \sum_{i=1}^n e_i \geq 0 \right\}. \quad (35)$$

As an example, if 8 sampling points are numbered from 1 to 8 and one T_x covers the sampling points 1, 2, 5, and 7, respectively, and another T_x covers the sampling points 2, 3, 7, and 8, respectively, then the coverage pattern for that T_x will be as in Table 1. Here, the resultant pattern C^* is created by merging both C_1 and C_2 based on the concept of logical inclusive "OR" operation, as shown in Table 1, where the result is "1," if the first bit is "1" OR the second bit is "1" or both bits are "1." Otherwise, the result is "0."

TABLE 1: Coverage pattern for 8 sampling points.

		e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
		↓	↓	↓	↓	↓	↓	↓	↓
C_1	=	1	1	0	0	1	0	1	0
C_2	=	0	1	1	0	0	0	1	1
C^*	=	1	1	1	0	1	0	1	1

The probability of T_x is based on a few principles.

- (i) If the number of e_i covered by the T_x becomes higher, then the term for summation will increase. Therefore, the probability of the T_x to be selected will become higher. This is named as intraprobability. Intraprobability for T_x is

$$t_i = \sum_{i=1}^n \text{multiplication term of } P_{ij}. \quad (36)$$

- (ii) If the numbers of T_x s which cover the same sampling point become smaller, then the probability of the T_x to be selected for that particular sampling point level will become higher. This is named as interprobability. Interprobability for each sampling point level is

$$P_{ij} = \frac{1}{\left(\sum_{i=1}^n t_{ie_j=1} \right)}. \quad (37)$$

- (iii) The probability for each T_x can be equal to 1, if and only if the T_x consists of a sampling point which is not covered by other T_x s; otherwise, the probability of the T_x should be always smaller than 1.
- (iv) If the probability is equal to 1, the integer value shows the number of sampling points that are only covered by this T_x and not other T_x s.
- (v) Only the maximum probability of each $T_x P_i$ will be taken.

For example, if there are 6 sampling points from 1 to 6, the coverage pattern of these six T_x s will be as shown in Table 2.

The first level sampling point, L_1 , is only covered by t_1 , t_2 , and t_3 ; therefore, the probability for L_1 , P_{11} , will be $1/3$. By using the same principle, the probability of L_2 , L_3 , L_4 , L_5 , and L_6 is obtained, as shown in Table 2. By using the above data, we found the probability for the t_1 as 0.74074 ($T_{P_1} = P_{13} + P_{13} * P_{11} + P_{13} * P_{11} * P_{12} + P_{13} * P_{11} * P_{12} * P_{14}$) and the probability for t_4 as 1.3333 ($T_{P_4} = P_{46} + P_{46} * P_{45}$).

Some of the bounding functions as well as termination criteria are applied while going deeper into the search space. For n number of sampling points in the indoor environment, the proposed bounding functions are as follows.

- (i) The probability of each existing and nonselected T_x for each sampling point level is recalculated, where the calculation starts from the current sampling point level up to n th level of sampling point. Therefore, the probability of each node will be reduced from sampling point level to level.

TABLE 2: Example of sampling pattern for 6 T_x s.

	e_1	e_2	e_3	e_4	e_5	e_6
$t_1 =$	1	1	1	1	0	0
$t_2 =$	0	1	0	1	0	0
$t_3 =$	1	0	0	0	1	0
$t_4 =$	0	0	0	0	1	1
$t_5 =$	1	1	0	0	0	0
$t_6 =$	0	0	1	1	1	0
P_{ij}	1/3	1/3	1/2	1/3	1/3	1

FIGURE 7: State space search tree generated by the proposed algorithm.

- (ii) The process needs to be run for $n + 1$ times, by removing one of the T_x s each time. The step is used to make sure the chosen T_x s are the minimum number of T_x s required to achieve the optimum wireless coverage. Besides, this step is also able to obtain another alternative for choosing T_x without degrading the coverage.

For making the proposed algorithm complete, the following termination criteria have been used. The termination happens if one of the following conditions becomes true.

- (i) The existing T_x in current sampling point level is one of the chosen T_x s in previous sampling point level. Termination happens to the current sampling point level and proceeds to the next level.
- (ii) Algorithm termination happens when no live node exists in the solution space to be explored.

To illustrate the proposed method, suppose there are 6 sampling points in the indoor propagation area and the coverage pattern is shown in Table 2. Figure 7 generates the state space search tree based on sampling point level to achieve the first optimal solution.

Initially, there is only one live node, node 0. This represents the case in which no T_x has been placed in the propagation area. This node becomes the E -node at the initial state. It is expanded based on sampling point level T_x ; therefore, for sampling point level, L_1 , its child nodes 1, 2, and 3 are generated. These nodes represent the solution space, where only one T_x is considered at a time. By calculating

the probability of t_1 , t_3 , and t_5 as shown in Table 2, the t_1 has a higher probability; therefore, the next E -node will be node 1 and the algorithm switches to the next sampling point level. Since, in sampling point levels 2, 3, and 4, t_1 exists again, according to the termination criteria, all these levels will be terminated and the E -node will remain as 1.

For 5th sampling point, only 3rd, 4th, and 5th T_x s exist in this sampling point level, by applying the first bounding function, where the probability of these T_x s will be recalculated, starting from the current sampling point level. Since sampling point level L_6 is only covered by 6th T_x , therefore, according to the third probability principle, the probability for 4th T_x is equal to 1. Since, in last sampling point level, t_4 exists again, the termination criteria will terminate this level and the E -node will remain as 5. Then, the resultant coverage pattern is as follows:

$$C^* = C_1 \cup C_4 = \boxed{1 \mid 1 \mid 1 \mid 1 \mid 1 \mid 1}. \quad (38)$$

The above pattern shows the primary optimum solution will be formed by the 1st T_x and 4th T_x . By applying the second bounding function, the proposed algorithm will be rerun again by removing one of the T_x s each time; therefore, the algorithm needs to run for 7 times in order to get the correct optimum solution by comparing the latest coverage pattern with the primary optimum solution obtained and also the number of T_x s required. Figure 7 is the first step and Figure 8 represents the rest of the steps.

3.1. Complexity Analysis. In this section, comparison between the proposed and existing algorithms [19, 20] is presented in terms of both space and time complexities. Here, the received signal at each sampling point due to one or more T_x s is calculated by using the ray tracing algorithm. Initially, for the worst case, each T_x covers one sampling point; therefore, n numbers of T_x s are needed for n number of sampling points. The ray tracing algorithm runs for every sampling point from the location of the T_x to generate the coverage pattern for each T_x . At the beginning, the signal is transmitted from a particular T_x situated at a certain location (e.g., A) to each of the sampling points. If the signal is able to be detected by another T_x situated at another particular sampling point (e.g., B), then the signal generated by the T_x of location B should also be able to reach the T_x located at A. Therefore, the number of ray tracing algorithms to be run decreases for each T_x . For example, if there are n numbers of sampling points, then 1st T_x is required to run ray-tracer $n - 1$ times and 2nd T_x will run $n - 2$ times, and so on. Therefore, the total amount of ray tracing algorithms to be run for n numbers of T_x s is

$$(n - 1) + (n - 2) + (n - 3) + \cdots + 3 + 2 + 1 = \sum_{i=1}^n n - i. \quad (39)$$

The proposed algorithm requires all T_x sampling pattern information in order to perform the probability calculation. Hence, before proceeding to the proposed algorithm, the ray tracing algorithm needs to run completely for generating

FIGURE 8: Different stages of getting optimum solution. T_x removed (a) R_1 , (b) R_2 , (c) R_3 , (d) R_4 , (e) R_5 , and (f) R_6 .

FIGURE 9: Tree organization for the proposed coverage algorithm.

sampling pattern for all T_x s. Based on this algorithm, for a typical indoor environment having 5 sampling points, the worst case search tree can be generated, as shown in Figure 9.

Node 0 refers to only the root node of the tree, so it does not refer to any T_x . Therefore, node 0 of the tree can be referred to as dummy node. By referring to the tree organization of Figure 9, the total space tree required is

$$\text{Total space tree} = 6 + 5 + 4 + 3 + 2 + 1 + \text{dummy node.} \quad (40)$$

It can be further simplified by writing the sum of the natural numbers up to n in two ways as

$$\begin{aligned} S_n &= 1 + 2 + 3 + \dots + (n-2) + (n-1) + n, \\ S_n &= n + (n-1) + (n-2) + \dots + 3 + 2 + 1. \end{aligned} \quad (41)$$

Adding these two series,

$$2S_n = (n+1) + (n+1) + \dots + (n+1). \quad (42)$$

There are n of these $(n+1)$'s, so

$$\begin{aligned} 2S_n &= n(n+1), \\ S_n &= \frac{n(n+1)}{2}. \end{aligned} \quad (43)$$

Hence, the sum of the natural numbers from 1 to n is therefore half the product of the first term plus the last one multiplied by the number of terms. Thus, in general, the total number of nodes generated by the tree can be calculated as the following formula:

$$\frac{n(n+1)}{2} + \text{dummy node} = \frac{n(n+1)}{2} + 1, \quad (44)$$

where n is the number of sampling points in the selected indoor environment. As all the generated nodes stay in

TABLE 3: Complexities of the existing and the proposed methods.

Complexity	Existing [20]	Existing [19]	Proposed
Space	2^n	$2^{n-m} - k$	$n(n+1)/2 + 1$
Time	2^n	$2^{n-m} - k$	$n(n+1)/2 + 1$

memory, therefore, the space complexity is also the same as the time complexity. The space complexity refers to the number of nodes generated until the deepest level, and the time complexity depends on the number of nodes generated until the required solution has been found [19].

The proposed algorithm will be compared with the existing algorithms [19, 20]. According to existing algorithm [20], the time complexity can be expressed as

$$\begin{aligned} &1 + \sum_{r=1}^n \sum_{i=1}^1 (r-i) + \sum_{r=2}^n \sum_{i=1}^2 (r-i) + \sum_{r=3}^n \sum_{i=1}^3 (r-i) \\ &+ \dots + \sum_{r=n-1}^n \sum_{i=1}^{n-1} (r-i) + \sum_{r=n}^n \sum_{i=1}^n (r-i) \\ &= 1 + \sum_{r=1}^n C_0 + \sum_{r=2}^n C_1 + \sum_{r=3}^n C_2 + \dots + \sum_{r=n-1}^n C_{n-1} + \sum_{r=n}^n C_n \\ &= 2^n, \end{aligned} \quad (45)$$

where r is the number of rows and n is the number of sampling points in the indoor environment. Equation (45) represents the time complexity of the existing algorithm [20] for the worst case.

The time complexity of the existing algorithm [19] can be expressed as follows by modifying existing algorithm [20]:

$$2^{n-m} - k, \quad (46)$$

where n is the number of sampling points in the selected indoor environment, m is the number of coverage patterns that has been rejected because of duplication, and k is the number of nodes that has been unexplored. In the worst case, the value of both m and k will be 0.

From Table 3, it is clear that both time and space complexities of the proposed algorithm are better than the existing algorithms [19, 20]. From the table, it is also apparent that the time and space complexities of the existing algorithms are increasing exponentially while the space and time complexity of the proposed algorithm increase linearly. Therefore, we can say that the proposed algorithm will take significantly lower execution time than the existing algorithms.

4. Results and Discussion

For fair comparison, the same environments are used and all experimental settings have been kept equivalent. All data in this section have been taken with the same simulation setup. All of the comparable techniques (in a format which gives the best possible outcome) have been implemented in the same simulation software. All simulations are carried

TABLE 4: Properties of different materials.

Material	Dielectric constant, ϵ_r	Refractive index, η
Brick	5.2	
Glass	3	1.52
Wood	3	
Plastic board (PVC)	4	1.46

FIGURE 10: Comparison in terms of time.

out on a PC consisting of Intel(R) Core(TM) 2 Duo CPU E8400 @ 3.00 GHz, 3.23 GB of RAM, operated by Microsoft Windows XP operating system. Thus, there is no chance of discriminations between the techniques. Though this proposed technique is applicable for any frequencies, in most of the cases, 2.4 GHz operating frequency and half wave dipole antenna with 2.15 dBi gain have been used. The objects of the simulation software were made by using cuboids. The properties of different materials are used to make the simulation environment the same as realistic environments. The material properties are shown in Table 4.

To evaluate the performance of the proposed technique, a comparison is made with the existing methods. The comparison is made between the proposed technique and the SBR, BT, BDPT, RF, PDM, and SD techniques. The drawbacks of the existing techniques have been described in Section 1. For proper comparison, five different environments are chosen (one of them is shown in Figure 5). The environments are different by means of a number of objects. Some are mostly complex and some are moderate. Measurements are done in 10 different sampling points for each environment, by changing the T_x and R_x positions. The results obtained from 10 different sampling points of Figure 5 are represented graphically in Figure 10. Table 5 represents the overall results for all five environments.

According to Figure 10, the proposed algorithm shows lower time consumption for the environment shown in Figure 5. Results for five such kinds of different environments are represented in Table 5. From the results, we observe that the proposed algorithm shows 69.93% lower time consumption than SBR algorithm, 63.67% lower than BT, 82.44% lower than BDPT, 66.03% lower than RF technique, 83.36% lower

than PDM method, and 82.12% lower than SD technique. This is because the DQS helps to decrease the time by skipping objects during simulation, the COF technique minimizes the ray-object intersection time, and the AVL tree data structure minimizes the data searching time.

For the justification of the inclusion of rough surface scattering and different optimization techniques, this part presents the effects of rough surface scattering and the proposed optimization techniques on prediction time. These effects are presented graphically in Figure 11 for different scattering factors (SF). SF is the key feature which has impact on scattering simulation. SF has a key impact on scattering angle. When the SF increases, the scattering angle also increases. Thus, the chance of ray-object interaction increases and it increases the prediction time. Here, a comparative study is presented to show the improvement of the proposed technique step-by-step. For $SF = 4$ (Figure 11(a)), after including the scattering in the ray tracing technique, the time decreases by 4.65% in average compared to the algorithm that has not considered scattering in ray tracing. Furthermore, 11.68% time reduction is achieved for the inclusion of DQS optimization technique and 31.55% reduction for the inclusion of both DQS and COF techniques. For $SF = 8$ (Figure 11(b)) and $SF = 20$ (Figure 11(c)), the time also decreases step-by-step after including the scattering and different optimization techniques. These results demonstrate strong effects of scattering and optimization techniques on prediction time.

As we know, there are a number of types of antennas, which can be used as T_x and R_x . For different types of antennas, the received power and electric field strength will be different due to different antenna characteristics (i.e., antenna gain). Figure 12 shows three different antennas and the electric field strength in the receiver side. Here, the antennas used are half wave dipole antenna with 2.15 dBi gain, bow-tie antenna with 2.3 dBi gain, and hemispherical antenna with 7 dBi gain. In all cases, 2.4 GHz operating frequency is used for data collection. In the figures, almost all of the techniques show nearly similar field strength at the receiver end. There are small differences between the techniques and this difference is a result of multipath propagation, which is represented by different approaches in different algorithms. As a result, variation occurs in the electric field strength at the receiver end. Another cause of the variation is different optimization techniques used in the algorithms. Due to these optimization techniques, the amount of the received signal varies and, thus, the electric field strength also varies. This nearly similar result of different techniques verifies the validity of the proposed technique.

In the above comparisons, verification is done by the data taken using different antennas with the same operating frequency. In this part, the same thing is done by changing the operating frequency for the same antenna to observe the effect of different frequencies. As it is known that the operating frequency has an impact on the path loss calculation [27], Figure 13 represents the path loss data for half wave dipole antenna in two different frequencies. 900 MHz frequency is used for Figure 13(a) and 2.4 GHz is used for Figure 13(b). For 900 MHz, the path loss varies from 38.8 dB to 45 dB

TABLE 5: Combined results for all five environments.

		Proposed	SBR	BT	BDPT	RF	PDM	SD
Time (ms)	1	112	407	352	528	365	646	521
	2	76	247	204	487	221	492	476
	3	69	231	192	473	211	479	461
	4	85	269	216	502	242	511	498
	5	105	328	265	545	275	551	534

(a)

— Without scattering — Scattering + DQS
— Scattering only — Scattering + DQS + COF

(b)

— Without scattering — Scattering + DQS
— Scattering only — Scattering + DQS + COF

(c)

— Without scattering — Scattering + DQS
— Scattering only — Scattering + DQS + COF

FIGURE 11: Effects of scattering and the proposed optimization techniques on time for (a) SF = 4, (b) SF = 8, and (c) SF = 20.

and for 2.4 GHz, it varies from 53.1 dB to 59 dB for different techniques. It indicates that, due to increasing the operating frequency, the path loss also increases.

In previous discussions, the validity and the superiority of the proposed ray tracing technique are done with the simulation results. However, in this part, the comparison between

the simulated and the experimental data is also shown in order to evaluate the validity of the proposed technique. In real cases, the transmitted power is constant and the received power is variable for different R_x s depending on the T_x - R_x separation, the amount of obstacles, and so forth. Based on the received power, the signal strength of R_x changes,

FIGURE 12: Electric field strength with (a) half wave dipole antenna, (b) bow-tie antenna, and (c) hemispherical antenna.

which makes the received signal a “good” signal or a “bad” signal. For this reason, the received power has significance in ray prediction and we have chosen this parameter for the comparison. Figure 14 shows the comparison between the simulated and experimental received power for different scenarios. During experiment and simulation, the transmitted power is kept constant. From the figure, it is observed that the simulation results are almost matching with the experimental results, which validate the proposed algorithm. The average difference between the simulation and experimental data is very small and it is 1.65%. These differences are the effects of different environmental parameters, object properties, and the existence of other electromagnetic signals in the practical environment.

Until now, it is apparently found that the proposed ray tracing technique is valid and more efficient than the existing ray tracing techniques. Here, validation of the proposed coverage algorithm is deliberated in comparison with the simulated and experimental results for different transmitted power. The coverage for a specific environment is changeable with the changes in transmitted power. If the transmitted power is constant, the coverage will depend on the obstacles. If the number of obstacles or position of obstacles changes, it will make changes in the coverage. Therefore, in case of constant transmitted power, the coverage will remain almost the same for the same environment. For this reason, we have verified our algorithm with different transmitted power for the same environment and achieved approximately 99%

FIGURE 13: Comparison between the proposed and existing algorithms based on the average path loss for (a) 900 MHz and (b) 2.44 GHz.

FIGURE 14: Received power in different scenarios.

FIGURE 15: Comparison between simulated and experimental coverage.

and 98% maximum coverage in terms of simulation and experiment, respectively, for 10 dBm transmitted power (refer to Figure 15). Also, from Figure 15, we observe the matching results between the simulation and experimental results of the coverage algorithm. The average difference between the simulation and experimental data is 0.3%. This difference is the result of different environmental effects and trivial mismatching between the simulation and experimental setup. These matching results confirm the accuracy of the proposed coverage algorithm.

5. Conclusion

A new propagation prediction technique for indoor environment is proposed in this paper where AVL tree is used for data storing and retrieving process, and DQS and COF techniques are used for accelerating the overall ray tracing process. This paper also proposed a new algorithm for wireless indoor coverage, which is based on the probability theory and optimized by GA and DFS. Analysis between the proposed

method and the existing methods proved that the proposed method has lower time and space complexities. The obtained results reveal that the proposed technique improved the performance in terms of lower computational time of about 83.36%. In case of coverage algorithm, the space and time complexities of the proposed algorithm are greatly improved because of strong bounding functions and termination criteria as well as the concept of multilevel technique for finding the suitable T_x s from the coverage pattern based on the probability. The similarities between the simulated and experimental results (a very high matching accuracy of more than 98% with respect to received power and coverage) confirm the validity of the proposed technique.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This research work is supported by the University of Malaya High Impact Research (HIR) Grant (UM.C/628/HIR/ENG/51) sponsored by the Ministry of Higher Education (MOHE), Malaysia.

References

- [1] P. Combeau, L. Aveneau, R. Vauzelle, and Y. Pousset, "Efficient 2-D ray-tracing method for narrow and wideband channel characterisation in microcellular configurations," *IEE Proceedings: Microwaves, Antennas and Propagation*, vol. 153, no. 6, pp. 502–509, 2006.
- [2] R. Hoppe, P. Wertz, F. M. Landstorfer, and G. Wölfle, "Advanced ray-optical wave propagation modelling for urban and indoor scenarios including wideband properties," *European Transactions on Telecommunications*, vol. 14, no. 1, pp. 61–69, 2003.
- [3] M. S. Sarker, A. W. Reza, and K. Dimyati, "A novel ray-tracing technique for indoor radio signal prediction," *Journal of Electromagnetic Waves and Applications*, vol. 25, no. 8-9, pp. 1179–1190, 2011.
- [4] Y. B. Tao, H. Lin, and H. J. Bao, "KD-tree based fast ray tracing for RCS prediction," *Progress in Electromagnetics Research*, vol. 81, pp. 329–341, 2008.
- [5] J. K. Bang, B. C. Kim, S. H. Suk, K. S. Jin, and H. T. Kim, "Time consumption reduction of ray tracing for RCS prediction using efficient grid division and space division algorithms," *Journal of Electromagnetic Waves and Applications*, vol. 21, no. 6, pp. 829–840, 2007.
- [6] A. W. Reza, K. Dimyati, K. A. Noordin, A. S. M. Z. Kausar, and M. S. Sarker, "A comprehensive study of optimization algorithm for wireless coverage in indoor area," *Optimization Letters*, vol. 8, pp. 145–157, 2012.
- [7] A. S. M. Z. Kausar, A. W. Reza, K. A. Noordin, M. J. Islam, and H. Ramiah, "Efficient radio propagation prediction algorithm including rough surface scattering with improved time complexity," *Progress In Electromagnetics Research B*, vol. 53, pp. 127–145, 2013.
- [8] N. S. Alvar, A. Ghorbani, and H. Amindavar, "A novel hybrid approach to ray tracing acceleration based on pre-processing & bounding volumes," *Progress in Electromagnetics Research*, vol. 82, pp. 19–32, 2008.
- [9] C. H. Teh, F. Kung, and H. T. Chuah, "A path-corrected wall model for ray-tracing propagation modeling," *Journal of Electromagnetic Waves and Applications*, vol. 20, no. 2, pp. 207–214, 2006.
- [10] M. F. Cátedra, J. Pérez, F. Saez de Adana, and O. Gutierrez, "Efficient ray-tracing techniques for three dimensional analyses of propagation in mobile communications: Application to picocell and microcell scenarios," *IEEE Antennas and Propagation Magazine*, vol. 40, no. 2, pp. 15–28, 1998.
- [11] K.-S. Jin, T.-I. Suh, S.-H. Suk, B.-C. Kim, and H.-T. Kim, "Fast ray tracing using a space-division algorithm for RCS prediction," *Journal of Electromagnetic Waves and Applications*, vol. 20, no. 1, pp. 119–126, 2006.
- [12] Y. Cocheril and R. Vauzelle, "A new ray-tracing based wave propagation model including rough surfaces scattering," *Progress in Electromagnetics Research*, vol. 75, pp. 357–381, 2007.
- [13] M. Thiel and K. Sarabandi, "A hybrid method for indoor wave propagation modeling," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 8, pp. 2703–2709, 2008.
- [14] H. Kim and H. Lee, "Accelerated three dimensional ray tracing techniques using ray frustums for wireless propagation models," *Progress in Electromagnetics Research*, vol. 96, pp. 21–36, 2009.
- [15] Z. Yun, M. F. Iskander, and Z. Zhang, "Fast ray tracing procedure using space division with uniform rectangular grid," *Electronics Letters*, vol. 36, no. 10, pp. 895–897, 2000.
- [16] Z. Xu, H. Li, Q.-Z. Liu, and J.-Y. Li, "Pattern synthesis of conformal antenna array by the hybrid genetic algorithm," *Progress in Electromagnetics Research*, vol. 79, pp. 75–90, 2008.
- [17] Z. Meng, "Autonomous genetic algorithm for functional optimization," *Progress in Electromagnetics Research*, vol. 72, pp. 253–268, 2007.
- [18] Y. A. Fadil, "Routing using genetic algorithm for large networks," *Journal of Engineering Sciences*, vol. 3, no. 2, pp. 52–70, 2010.
- [19] A. W. Reza, M. S. Sarker, and K. Dimyati, "A novel integrated mathematical approach of ray-tracing and genetic algorithm for optimizing indoor wireless coverage," *Progress in Electromagnetics Research*, vol. 110, pp. 147–162, 2010.
- [20] Z. Yun, S. Lim, and M. F. Iskander, "An integrated method of ray tracing and genetic algorithm for optimizing coverage in indoor wireless networks," *IEEE Antennas and Wireless Propagation Letters*, vol. 7, pp. 145–148, 2008.
- [21] D. E. Knuth, *The Art of Computer Programming*, Addison-Wesley, London, UK, 3rd edition, 1997.
- [22] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, The MIT Press, Cambridge, Mass, USA, 2nd edition, 2001.
- [23] J. A. Ogilvy, *Theory of Wave Scattering from Random Rough Surfaces*, Institute of Physics, 1991.
- [24] V. Mohtashami and A. A. Shishegar, "Efficient shooting and bouncing ray tracing using decomposition of wavefronts," *IET Microwaves, Antennas & Propagation*, vol. 4, no. 10, pp. 1567–1574, 2010.

- [25] M. F. Iskander and Z. Yun, "Propagation prediction models for wireless communication systems," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 3, pp. 662–673, 2002.
- [26] N. Herscovici, C. Christodoulou, F. Gil et al., "A 3D interpolation method for base-station-antenna radiation patterns," *IEEE Antennas and Propagation Magazine*, vol. 43, no. 2, pp. 132–137, 2001.
- [27] M. Riback, J. Medbo, J. E. Berg, F. Harrysson, and H. Asplund, "Carrier frequency effects on path loss," in *Proceedings of the Vehicular Technology Conference*, May 2006.