Global Optimization of Transmitter Placement in Wireless Communication Systems

J. He*, A. Verstak*, L. T. Watson*, T. S. Rappaport†, C. R. Anderson†, N. Ramakrishnan*, C. A. Shaffer*, W. H. Tranter†, K. Bae†, J. Jiang†
Department of Computer Science*
Bradley Department of Electrical and Computer Engineering†
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061
Contact E-mail: jihe@vt.edu

Keywords: DIRECT algorithm, global optimization, parallel ray tracing, pattern search, transmitter placement.

Abstract

This paper explores the application of a global optimization technique to solve the optimal transmitter placement problem in wireless system design. An efficient pattern search algorithm—DIRECT (DIviding RECTangles) of Jones, Perttunen, and Stuckman (1993)—has been connected to a parallel 3D radio propagation ray tracing modeler running on a Beowulf cluster of workstations. The algorithm optimizes, for a given computational investment, the locations of a specified number of transmitters across the feasible region of the design space. The focus of the paper is on the implementations of the DIRECT algorithm and the parallel 3D ray tracing propagation model. Both simulation results and site measurement data are presented in support of the effectiveness of the present work.

1. INTRODUCTION

Optimal transmitter placement provides high spectral efficiency and system capacity while reducing network costs, which are the key criteria for wireless network planning. As the complexity and popularity of modern wireless networks increases, automatic transmitter placement provides cost savings when compared to the traditional human process of site planning. Automatic design tools are being developed to offer efficient and optimal planning solutions. Besides [4] and [7], S^4W (Site-Specific System Simulator for Wireless system design) is among the few known wireless system tools for in-building network design. It is being developed jointly by the Mobile & Portable Radio Research Group (MPRG) and the Problem Solving Environment (PSE) research group at Virginia Polytechnic Institute & State University. An optimization loop in S^4W is proposed to maximize the efficiency of simulated channel models and surrogate functions are proposed to reduce the cost of simulations. Transmitter placement optimization is one specific problem that can be solved by S^4W. An example of the S^4W system model consisting of a propagation model, a channel model, and an optimizer is given in [15].

The underlying optimization algorithm is known as DIRECT (DIviding RECTangles), a direct search algorithm proposed by Jones et al. [8]. It was proposed as an effective approach to solve global optimization problems (GOP) subject to simple constraints. Jones et al. [8] named the algorithm after one of its key steps—dividing rectangles. DIRECT is a pattern search method that is categorized as a direct search technique by Lewis et al. [9]. Generally speaking, "pattern search methods are characterized by a series of exploratory moves that consider the behavior of the objective function at a pattern of points" [9], which are chosen as the centers of rectangles in the DIRECT algorithm. This center-sampling strategy reduces the computational complexity, especially for higher dimensional problems. Moreover, DIRECT adopts a strategy of balancing local and global search by selecting potentially optimal rectangles to be further explored. This strategy gives rise to fast convergence with reasonably broad space coverage. So far, DIRECT has been used with fair success for modern large-scale multidisciplinary engineering problems [1]. The present work is the first known application to wireless communication systems.

In general, transmitter placement optimization is aimed at covering a geographical area of interest to a specified minimum power level (threshold) at a minimum cost [4]. In [4], "coverage" is defined as the ratio of the number of receiver locations with received power above the threshold to the total number of receiver locations. This nonsmooth function leads to the rank based methods used by [4]. In [7] and [12], the objective function is based on several weighted factors, such as covered

area, interference area, and mean signal path loss. In the present work, the objective function is a continuous penalty function devised to minimize the shortfall of received power with respect to the power threshold. 3D ray tracing is used as a deterministic propagation model to estimate coverage levels for transmitter locations sampled by DIRECT. This site-specific technique is widely used for simulating wireless channels [13],[14]. It provides both small-scale and large-scale propagation information which enables its use in a variety of wireless design and deployment scenarios. Since 3D ray tracing is computationally expensive, an MPI-based parallel implementation is used in the present work.

This paper is organized as follows. An overview of the DIRECT algorithm is given in Section 2, followed by a description of dynamic data structures. Section 3 presents the parallel 3D ray tracing model. In Section 4, optimization results are analyzed. Finally, Section 5 summarizes some key contributions of the present work and suggests directions for future research.

2. DIRECT

The multivariate DIRECT algorithm can be described by the following six steps [8].

Given an objective function f and the design space $D = \{x \in E^n \mid \ell \leq x \leq u\}$:

Step 1. Normalize the design space D to be the unit hypercube. Sample the center point c_i of this hypercube and evaluate $f(c_i)$. Initialize $f_{\min} = f(c_i)$, evaluation counter $m = 1$, and iteration counter $t = 0$.

Step 2. Identify the set S of potentially optimal boxes.

Step 3. Select any box $j \in S$.

Step 4. Divide the box j as follows:

(1) Identify the set I of dimensions with the maximum side length. Let δ equal one-third of this maximum side length.

(2) Sample the function at the points $c \pm \delta e_i$ for all $i \in I$, where c is the center of the box and e_i is the ith unit vector.

(3) Divide the box j containing c into thirds along the dimensions in I, starting with the dimension with the lowest value of $w_i = \min\{f(c+\delta e_i), f(c-\delta e_i)\}$, and continuing to the dimension with the highest w_i. Update $f_{\min}$ and m.

Step 5. Set $S = S - \{j\}$. If $S \neq \emptyset$ go to Step 3.

Step 6. Set $t = t + 1$. If iteration limit or evaluation limit has been reached, stop. Otherwise, go to Step 2.

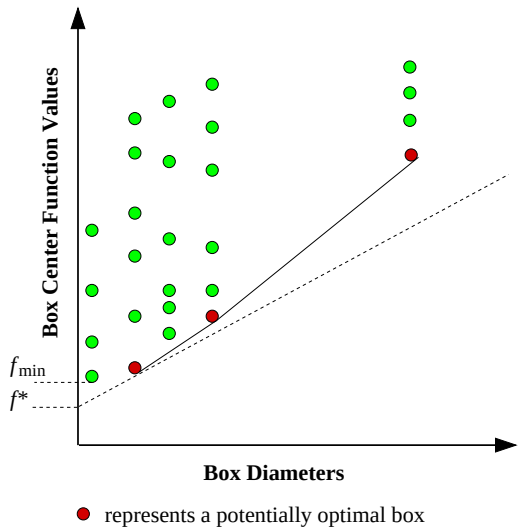

Figure 2.1. Illustration of potentially optimal boxes on convex hull with ϵ test. Note that $f^* = f_{\min} - \epsilon|f_{\min}|$. Potentially optimal boxes are on the lower-right convex hull.

Steps 2 to 6 form a processing loop controlled by two stopping criteria —limits on iterations and function evaluations. Starting from the center of the initial hypercube, DIRECT makes exploratory moves across the design space by probing potentially optimal subsets. "Potentially optimal" is an important concept defined next [8].

Definition 2.1. Suppose that the unit hypercube has been partitioned into m (hyper) boxes. Let c_i denote the center point of the ith box, and let d_i denote the distance from the center point to the vertices. Let $\epsilon > 0$ be a positive constant. A box j is said to be *potentially optimal* if there exists some $\tilde{K} > 0$ such that for all $i = 1, \ldots, m$,

$$f(c_j) - \tilde{K} d_j \leq f(c_i) - \tilde{K} d_i, \tag{2.1}$$

$$f(c_j) - \tilde{K} d_j \leq f_{\min} - \epsilon|f_{\min}|. \tag{2.2}$$

Figure 2.1 represents the set of boxes as points in a plane. The first inequality (2.1) screens out the boxes that are not on the lower right of the convex hull of the plotted points, as shown in Figure 2.1. Note that $\tilde{K}$ plays the role of the (unknown) Lipschitz constant. The second inequality (2.2) prevents the search from becoming too local and ensures that a nontrivial improvement will (potentially) be found based on the current best solution. In Figure 2.1, $f_{\min}$ is the current best solution, but its associated box is screened out of the potentially optimal box set due to the second inequality (2.2).

Some modifications with respect to the stopping rules and box selection rules are proposed in the present

implementation to offer more choices. Two new stopping criteria are (1) minimum diameter (terminate when the best potentially optimal box's diameter is less than this minimum diameter) and (2) objective function convergence tolerance (exit when the objective function does not decrease sufficiently between iterations). The minimum diameter of a hyperbox represents the degree of space partition, and therefore is a reasonable criterion for optimization problems requiring only some depth of design space exploration. The objective function convergence tolerance was inspired by some experimental observations in the later stages of running the DIRECT algorithm, when the objective function convergence tolerance test avoids wasting a great number of expensive function evaluations in pursuit of very small improvements.

The present implementation of the DIRECT algorithm addresses an efficiency issue involved in an unpredictable storage requirement in the phase of space partitioning. To reduce the execution overhead and adapt to varying memory requirements, a set of dynamic data structures is proposed. They are extensible and flexible in dealing with information generated by the space partitioning process in high dimensions.

Two groups of dynamic structures have been implemented in Fortran 90: box structures and linked list structures illustrated by Figure 2.2. The box structures (`BoxMatrix`, `BoxLink`, and `HyperBox`) are responsible for holding boxes. The linked lists (`setInd`, `setDia`, and `setFcol`) are built out of linked vectors (`real_vector` and `int_vector`), and manage the allocated memory for the box structures.

In [8], Graham's scan method is recommended for finding the convex hull of a set of m arbitrary points in time $O(m \log_2 m)$. Here, a different approach is taken to shrink the initial set with m points to a much smaller set of vertices exclusively around the low edge of the convex hull. With all the hyperboxes linked logically in the scatter plot pattern, Jarvis's march (or gift wrapping) method is applied starting from the box sequence with the biggest size, and eventually identifies all the potentially optimal boxes to be further subdivided for the next iteration.

The linked list data structures play an important role in maintaining the logical scatter plot pattern and recycling memory cells. They are doubly linked lists constructed with two derived data types. In some sense, `M` (the two-dimensional array defined in `BoxMatrix`) acts as a memory pool of recyclable cells for holding boxes. When cells are used up, a new `BoxMatrix` is allocated and connected as the `child` link at the end of the chain of `BoxMatrices`, so that the memory pool can be filled up again using new cells from `M` in the newly

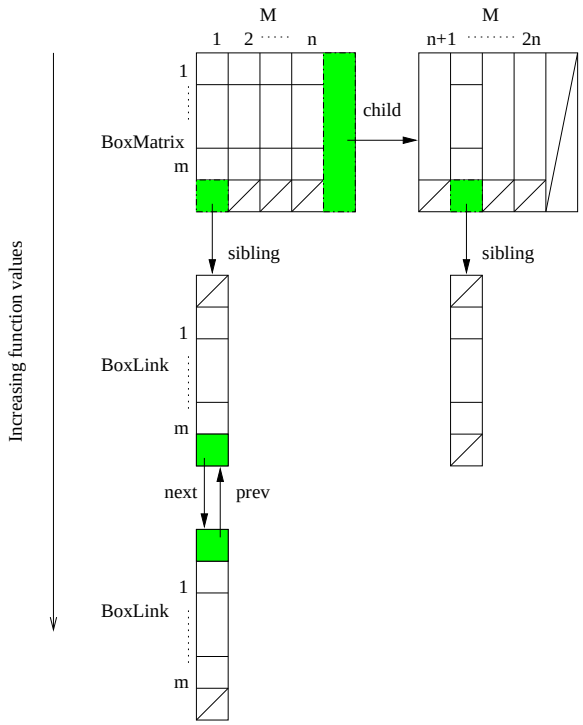

Figure 2.2. Box structures comprised of `HyperBoxes`.

allocated `BoxMatrix`. For faster execution, sorting is not involved in the strategy for maintaining a logical scatter plot pattern of hyperboxes. Instead, binary search is used in locating the insertion positions in sorted sequences, in both the cases of inserting boxes and box sizes. Some shifting operations are needed for inserting/deleting boxes in a particular column of boxes in `M` and its box links, if any, while shifting boxes among columns is avoided by keeping column indices sorted (by decreasing box sizes) in `setInd`.

3. PARALLEL RAY TRACING

Received power levels are approximated with a 3D ray tracing propagation model that is based on geometrical optics. Electromagnetic waves are modeled as rays that are traced through reflections and transmissions through the walls. Beam tracing is combined with ray tracing to improve simulation accuracy [3]. Beams are shot from geodesic domes drawn around transmitters. Each beam is a triangular pyramid formed by the point location of the transmitter and one of the triangles on the surface of the dome. Essentially, the spherical wavefront is triangulated and the 3D sphere is split into pyramidal beams. Following the argument in [14], all such beams are disjoint and have nearly the same shape and angular separation. Only the central ray of each beam is traced to identify reflection locations. However,

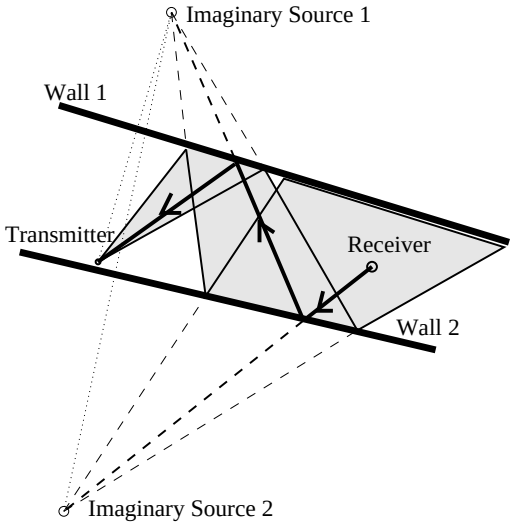

Figure 3.1. 2D beam tracing: a beam (shadowed region) is traced from the transmitter location to the receiver location through two reflections, and then a ray (bold line) is traced back.

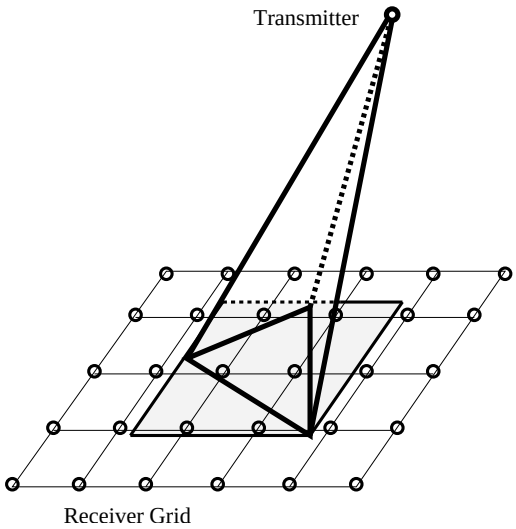

Figure 3.2. Beam intersection with a receiver grid: only the locations inside of the bounding box of the projection of the beam onto the grid (shadowed region) are tested for intersection with the beam pyramid.

the whole beam is used for ray-receiver intersection tests. Once an intersection with a receiver location is detected, a ray will be traced back from the receiver to the transmitter through the sequence of reflections and transmissions (penetrations) encountered by the beam. The illustration of this process in 2D is given in Figure 3.1. Neither diffraction nor scattering are modeled for computational complexity reasons, although these phenomena play an important role in propagation [11].

The goal of this combination of ray tracing and beam tracing is to keep the geometry simple and fast while eliminating double counting of the rays and avoiding local averaging of received signal levels. Double counting increases the communication cost of a parallel implementation since more rays need to be collected by the simulation. Also, beam tracing allows for fast computation of the power levels of uniformly placed receiver locations. In addition to point receivers, the ray tracer supports grids of receiver locations with uniform elevation and X and Y separation. To calculate beam intersections with all receiver locations on the grid, the ray tracer only considers the locations inside of the projection of the beam onto the grid, as shown in Figure 3.2. The number of such receiver locations is usually small compared to the total number of receiver locations on the grid.

Although material parameters and incidence angles affect losses in a wireless channel, a constant 6 dB reflection loss (same as in [13]) and a constant 4.6 dB transmission (penetration) loss (the loss for plaster board in [2]) are assumed. The power contribution of each ray, in dBW, is calculated according to the model developed in [14]:

$$P_j = P(d_0) - 20\log_{10}(d/\lambda) - nL_r - mL_t, \qquad (3.1)$$

where P_j is the power of the j-th ray, d is the total distance traveled by the ray, $P(d_0)$ is the transmitter power at a reference distance d_0 from the transmitter, n and m are the numbers of reflections and transmissions, $L_r = 6$ dB and $L_t = 4.6$ dB are reflection and transmission losses, and λ is the wavelength.

Two techniques are used to improve software performance. Classical octree space partitioning [6] reduces the number of wall intersection tests to an expected $O(\log N)$ per reflection, where N is the number of walls. A variation of image parallelism with dynamic scheduling [5] is applied to distribute the computation across multiple processors. Each processor has a complete copy of the building database broadcast by the master processor at the beginning of the computation. Beams are distributed by the master processor dynamically with an exponentially decreasing beam bundle size. This strategy achieves excellent load balancing at the cost of exchanging $O(\log T)$ messages between each processor and the master processor, where T is the geodesic tessellation frequency. With a large enough number of processors (up to 75 were used in simulations), collecting the powers at all receiver locations dominates both computation and other communication costs.

The ray tracer has been validated and calibrated with a series of measurements in the corridor of the fourth floor of Durham Hall, Virginia Tech. An ultra-wideband sliding correlator channel sounder operating

at 2.5 GHz and outfitted with omnidirectional antennas was used to record power delay profiles (PDPs) at six separate locations. The sliding correlator utilized an 11-bit, 400 MHz pseudo-noise spreading code for a time domain multipath resolution of 2.5 nanoseconds and a dynamic range of 30 dB. Simulated power delay profiles were post-processed and compared to the measured ones location by location.

The received E-field envelope of ray j (in V/m) that arrived at time t_j is $E_j = \sqrt{\eta 10^{0.1P_j}}$, where P_j is the output of the ray tracer (in dBW) and $\eta = 120\pi\ \Omega$ is the impedance of free space [11]. To account for antenna directivity, an omnidirectional antenna pattern must be applied to all E_js. The electric field that would be registered at time t_j by a hypothetical measurement system with infinite bandwidth resolution is

$$E'_j = E_j G_t G_r \cos\Theta_t \cos\Theta_r, \tag{3.2}$$

where Θ_t and Θ_r are ray transmission and reception elevation angles relative to the horizon, and G_t and G_r are maximum transmitter and receiver antenna gains, respectively. Since the measurement system had 2.5 ns time domain resolution, rays were split into bins of equal width $\delta = 2.5$ ns. The measured electric field

$$E_k^m = \sum_{j=1}^{Q} E'_j e^{-(t_j - k\delta)^2/(2\pi\sigma^2)} e^{i\phi_j} \tag{3.3}$$

of bin k centered at time $k\delta$ is a convolution of the predicted PDP (E'_js) with a Gaussian pulse transmitted by the channel sounder, where Q is the number of rays and $3\sigma = \delta$ is the time duration that contains 99% of the energy in the Gaussian pulse. In other words, every bin registers a weighted average of the energies of all predicted rays, where the weight decreases exponentially as the time difference of the ray and the bin increases. The complex factor $e^{i\phi_j}$ accounts for ray interference. Phase angles ϕ_j were determined from transmitter wavelength λ, total ray path length d_j, and number of reflections n (a 180 degree phase shift per reflection was assumed). Finally, $P_k^m = |E_k^m|^2/\eta$ gives the measured power of bin k, in watts.

Figure 3.3 shows measurements and predictions for one location with relatively strong multipath. As can be seen from the graph, the predictions are within 3–5 dB of the measurements, which is similar to the results achieved by earlier research [14]. The difference can be explained by device positioning errors (devices were positioned with ±3 cm precision, which is crude given that the wavelength was 12 cm) and imprecise modeling of reflections. Additionally, small multipath components were missed by the ray tracer. These components are probably due to scattering and diffraction, which were not simulated. Geodesic tessellation frequency was 700 (9.8×10^6 beams) for calibration because the simulation results for frequencies above 700 were indistinguishable.

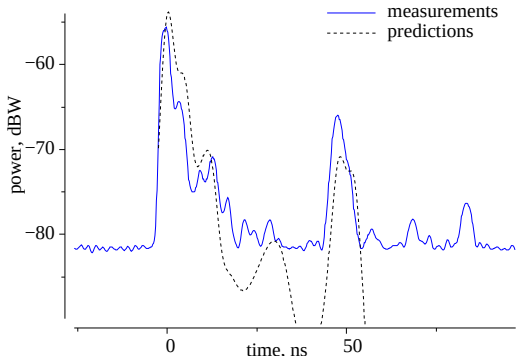

Figure 3.3. Measurement vs. prediction of channel impulse response.

4. OPTIMIZATION RESULTS

Ray tracing simulation was performed using only 40 nodes of a 200-node Beowulf cluster, because a large number of receiver locations generated considerable network traffic. Only peak powers from each receiver PDP were collected. Tessellation frequency was reduced from 700 to 100, because 100 was sufficient for matching peak powers. The optimizer interface ran on a Sun workstation outside the cluster. Tcl/Tk scripts glued the pieces together and provided a graphical user interface. Similar to [10], users could select regions for transmitter placement (to be optimized) and regions to be covered.

Consider the placement of n transmitters in an indoor environment located on the fourth floor of Durham Hall at Virginia Tech (see Figure 4.1). The variables are the transmitter coordinates

$$X = (x_1, y_1, z_1, x_2, y_2, z_2, \ldots, x_n, y_n, z_n),$$

where all $z_i = z_0$ are fixed, a reasonable assumption for indoor environments. The objective function is the average shortfall of the received peak power from the threshold, given by

$$f(X) = \frac{1}{m} \sum_{i=1}^{m} \min_{1 \le j \le n} (T - P_i(x_j, y_j, z_0))_+, \tag{4.1}$$

where T is the given power threshold (in dBm), P_i is the peak power received at the ith receiver (in dBm) from the transmitter located at (x_j, y_j, z_0), and m is the total number of receivers. Assume that transmitters operate at sufficiently different frequencies and that receivers pick up the strongest signal. $f = 0$ corresponds to perfect coverage. Optimizing a single transmitter location took 41 evaluations (3 minutes, 45 seconds) to reduce the objective function by 22.2% (from 4.60 dB to 3.58 dB) demonstrating the fast convergence of the DIRECT algorithm.

Figure 4.1 illustrates optimization of the locations of three transmitters to cover eighteen rooms and a corridor bounded by the box in the upper-left corner. 93 function evaluations reduced the objective from 2.77 dB to 2.51 dB, or by 9.4%, in 38 minutes and 20 seconds.

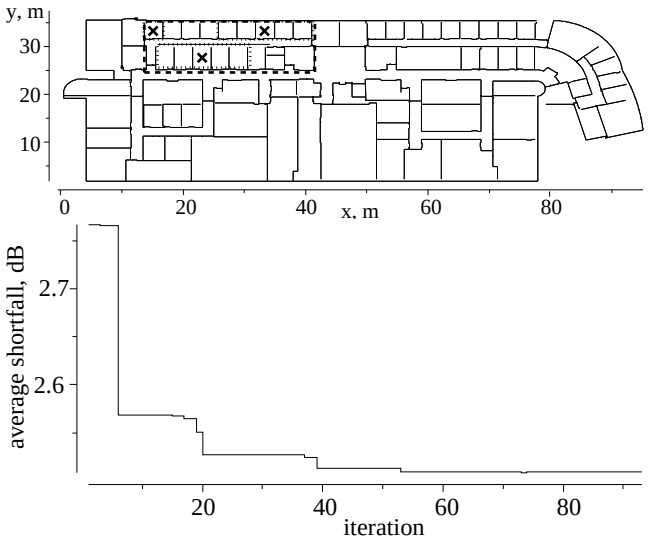

Figure 4.1 Optimization results for three transmitters. Bounds on transmitter placement are drawn with dotted lines and their initial positions are marked with crosses.

5. CONCLUSION

The main contribution of the present work is the design of the interface of a serial implementation of the DIRECT algorithm with a parallel formulation of a 3D ray tracing propagation model. DIRECT has demonstrated its effectiveness in solving the global optimization problem of transmitter placement in wireless communication systems design.

Several extensions to the present work are envisioned. First, computationally expensive ray tracing can be replaced by less accurate, but faster, propagation models. Such a formulation will allow for more function evaluations and require parallelization of DIRECT. Second, ray tracing output can be used to build sophisticated channel models that predict wireless system parameters such as bit error rate or bandwidth. Bit error rate simulations are extremely expensive, so surrogate functions are essential to make bit error rate optimization practical.

6. ACKNOWLEDGMENTS

This work was supported in part by NASA Grant NAG-2-1180, NSF Grant DMI-9979711, and NSF Grant EIA-9974956.

References

1. C. A. Baker, L. T. Watson, B. Grossman, R. T. Haftka, and W. H. Mason, "Parallel global aircraft configuration design space exploration", in Proc. High Performance Computing Symposium 2000, A. Tentner (Ed.), Soc. for Computer Simulation Internat., San Diego, CA, pp. 101–106, 2000.
2. G. D. Durgin, T. S. Rappaport, and Hao Xu, "Measurements and models for radio path loss and penetration loss in and around homes and trees at 5.85 GHz", IEEE Transactions on Communications, vol. 46(11), pp. 1484–1496, 1998.
3. S. J. Fortune, "A beam-tracing algorithm for prediction of indoor radio propagation", WACG: 1st Workshop on Applied Computational Geometry: Towards Geometric Engineering, WACG, LNCS.vol. 1148, pp. 157–166, 1996
4. S. J. Fortune, D. M. Gay, B. W. Kernighan, O. Landron, R. A. Valenzuela, and M. H. Wright (AT&T Bell Laboratories), "WISE design of indoor wireless systems: practical computation and optimization", IEEE Computational Science & Engineering, vol. 2(1), pp. 58–68, Spring, 1995.
5. B. Freisleben and D. Hartmann and T. Kielmann, "Parallel raytracing: a case study on partitioning and scheduling on workstation clusters", in Proc. Thirtieth International Conference on System Sciences, Hawaii, vol. 1, pp. 596–605, 1997.
6. A. Glassner, "Space subdivision for fast ray tracing", IEEE Computer Graphics and Applications, vol. 4(10), pp. 15–22, October, 1984.
7. X. Huang, U. Behr, and W. Wiesbeck, "Automatic base station placement and dimensioning for mobile network planning", in Proc. of Vehicular Technology Conference, IEEE VTS Fall VTC 2000. 52nd, vol. 4, pp. 1544–1549, 2000.
8. D. R. Jones, C. D. Perttunen, and B. E. Stuckman, "Lipschitzian optimization without the Lipschitz constant", Journal of Optimization Theory and Applications, vol. 79(1), pp. 157–181, 1993.
9. R. M. Lewis, V. Torczon, and M. W. Trosset, "Direct search methods: then and now", Journal of Computational and Applied Mathematics, vol. 124, pp. 191–207, 2000.
10. M. A. Panjwani, A. L. Abott, and T. S. Rappaport, "Interactive computation of coverage regions for wireless communication in multifloored indoor environments", IEEE Journal on Selected Areas in Communications, vol. 14(3), pp. 420–430, 1996.
11. T. S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall, 1996.
12. H. D. Sherali, C. M. Pendynala, and T. S. Rappaport, "Optimal location of transmitters for micro-cellular radio communication system design", IEEE Journal on Selected Areas in Communications, vol. 14(4), pp. 662–673, 1996.
13. K. R. Schaubach, N. J. Davis IV, and T. Rappaport, "A ray tracing method for predicting path loss and delay spread in microcellular environments", in Proc. IEEE Vehicular Technology Conference, vol. 2, pp. 932–935, 1992.
14. S. Y. Seidel and T. S. Rappaport, "Site-specific propagation prediction for wireless in-building personal communication system design", IEEE Transactions on Vehicular Technology, vol. 43(4), pp. 879–891, 1994.
15. A. Verstak, M. Vass, N. Ramakrishnan, C. Shaffer, L. T. Watson, K. K. Bae, J. Jiang, W. H. Tranter, and T. S. Rappaport, "Lightweight data management for compositional modeling in problem solving environments", in Proc. High Performance Computing Symposium 2001, A. Tentner (ed.), Soc. for Modeling and Simulation Internat., San Diego, CA, pp. 148–153, 2001.