

# FAST COMPUTATIONAL TECHNIQUES FOR INDOOR RADIO CHANNEL ESTIMATION

*When implemented on a parallel machine, the algorithm the authors have developed for channel estimation can significantly reduce execution time.*

**D**riven by our increasingly mobile society's ever-growing demand for communications, today's providers of wireless telecommunication services must ensure reliable radio coverage "everywhere." Computer-based techniques that reduce the need for expensive experimental measurements are invaluable tools for achieving this objective. Various computational algorithms based mainly on ray tracing have emerged in recent years for determining radio coverage (see the "Ray tracing" sidebar). Although the output of these algorithms agrees well with measurement results, execution times remain high.

To provide faster computational methods for determining radio coverage, we have developed a fast 3D method of regions (MR) algorithm (also described in the "Ray tracing" sidebar). This algorithm divides the environment into reflection, transmission, and possibly diffraction regions, then identifies infeasible image combinations and locations affected by similar propa-

gation mechanisms a priori. As this article shows, when implemented on a parallel machine, our algorithm provides close-to-ideal speedups, enabling the fast computation of radio coverage that service providers require.

## Method of Regions

Given a 3D scene, such as an office room, with a specified location for a wireless base station or transmitter, our MR algorithm computes a 3D-coverage map. This map determines the transmitter impulse response at each grid point  $(x, y, z)$  in the 3D scene. The algorithm thus can compute the optimal location for each wireless base station to maximize coverage. It considers three propagation modes:

- signal reflections due to panels,
- signal transmissions through panels, and
- signal diffractions due to panel edges.

This algorithm determines 3D regions, where all points in each region share nearly identical reflection, transmission, and diffraction effects, and where the regions are convex polyhedra. Unlike other approaches, it computes a 3D scene's true 3D-coverage map, rather than combining 2D results. Our MR algorithm can be considerably faster than the usual method of images (MI) algorithm, typically requiring one or

1521-9615/99/\$10.00 © 1999 IEEE

MARC KIMPE

*Adtran, Inc.*

HARRY LEIB, OLIVIER MAQUELIN, AND TED H. SZYMANSKI

*McGill University*

two orders of magnitude fewer computations. This algorithm is also inherently parallel, thus allowing fast parallel computation of true 3D-coverage maps.

In adopting a pedagogical approach, we'll use 2D figures to explain this algorithm's concept,

after which we'll present the generalization to three dimensions.

### Reflection

When a ray, representing an electromagnetic wave, intersects a smooth surface, two new rays

## Ray tracing

This decade has witnessed growing interest in wireless telecommunications for personal communications services.<sup>1</sup> Such systems allow tetherless access to a variety of telecommunication services such as transmission and reception of voice, image, data, and video signals. Successful deployment of PCS systems necessitates ubiquitous radio coverage, ensuring good quality of service for users regardless of their location. Consequently, ongoing research of PCS systems worldwide pays particular attention to radio systems aspects and propagation issues, including measurements and modeling techniques.

The countless obstacles to electromagnetic wave propagation—walls, for example—in both outdoor and indoor radio environments make predictions extremely difficult.

These obstacles scatter the electromagnetic waves and create multiple transmission paths between the transmitter and receiver. From a systems point of view, we can model the indoor radio environment as a fading multipath channel with baseband equivalent impulse response,

$$h(t) = \sum_i a_i e^{j\theta_i} \delta(t - \tau_i) \quad (1)$$

that represents the effect of different propagation paths from a transmitter to a receiver. Each path includes an attenuation  $a_i$ , carrier phase shift  $\theta_i$ , and delay  $\tau_i$ .<sup>2</sup> For determining radio coverage, we must estimate all or part of these parameters, then use these estimates to predict the channel quality.

Initially, the parameters  $a_i$ ,  $\theta_i$ , and  $\tau_i$  were statistically characterized, resulting in statistical models for the indoor radio channel. Such models can provide typical examples of channel-impulse responses but cannot predict the impulse response associated with a particular location. For this, we need site-specific predictions that associate  $a_i$ ,  $\theta_i$ , and  $\tau_i$  with a specific point in space. To have such predictions, we must estimate the parameters of Equation 1 typically for hundreds or thousands of locations. Future prediction tools likely will contain a site-specific component accounting for an environment's large and static elements (its walls, for instance) and a statistical component accounting for the small or dynamic elements (chairs and people, for example).

Site-specific indoor radio-channel estimation is essential for wireless telecommunications, thus combining the disciplines of electromagnetics, computer science, and computational

geometry. Solutions put forward by different researchers have the imprint of their original discipline. While electromagnetic purists will model as many modes and quirks of electromagnetic propagation as possible, computational scientists will look for elegant algorithms and culling techniques to solve the problem. Telecommunications engineers, however, must still determine which solutions will work in practice.

The application of ray-tracing techniques to channel estimation has grown significantly from its infancy in 1991, closely following the exponential growth in computational power. Ray tracing first found application in outdoor urban environments with low-rise buildings, where the problem is essentially 2D. Adding ground-reflection effects to the 2D solution resulted in "2.5D" solutions. The research focused on indoor estimation and evolved in three steps:

- (1) System developers first applied various ray-tracing algorithms to narrowband and then to wideband radio-channel estimation.<sup>3-7</sup> Nearly all methods generated estimates that correlated well with measurement results. The algorithms fall broadly into two classes: those relying on *ray shooting*, where a transmitter shoots rays in all directions, and those relying on the *method of images* (MI), which associates virtual sources with every obstacle and gauges their effects on transmission. Although the latter methods deliver a better tradeoff between prediction accuracy and execution speed, these methods are still too slow for use in a practical tool.
- (2) System developers developed faster algorithms to reduce computational time.<sup>8-12</sup>
- (3) They attempted to integrate channel estimation into a base-system developers station position-optimization tool that can identify locations for transmitters in a specific environment for ubiquitous radio coverage.<sup>9,13</sup>

The advent of wireless LANs has prompted the application of ray-tracing techniques to higher-frequency bands, extending the results to the 5.2-, 17-, and 20-to-60-GHz bands.<sup>14,15</sup> Although ray tracing seems to be the most popular method, it is not the only one for estimating coverage. Researchers have also proposed computational techniques based on the direct solution of Maxwell's equations: Finite-Difference Time Domain techniques (FDTD), Rayleigh-Gans approximation, and Transmission Line Matrix (TLM), for example.<sup>16-19</sup>

are generated: a reflected ray and a transmitted one. From the ray approximation of the Geometrical Theory of Diffraction and the theory of images,<sup>1</sup> we have that all reflections from a point source in a plane seem to originate from another fictitious source, called the *image source*.

The image source sits symmetrically on the other side of the plane along its perpendicular. We can construct multiple reflections by considering the image as a point source and continuing the image-creation process.

Only points in a specific spatial region lead to a

Another approach relying on path-loss computation has resulted in an efficient base-station optimization tool.<sup>13,20</sup> This tool creates radial paths outward from a base-station location and uses a path-loss model along each path. It then links the points of equal strength to generate contour maps.

Significant work during the last decade has gone into characterizing various indoor electromagnetic materials and propagation modes. The characterization of the channel's dynamic part resulted in a model representing pedestrian traffic using dielectric blocks.<sup>21</sup>

Most fast ray-tracing algorithms rely on spatial coherence. Two different approaches, unfortunately, share the same term—*beam tracing*. Strictly speaking, beam tracing is a ray-shooting algorithm in which many rays aggregate to form a beam to avoid missing objects and lower the number of ray queries.<sup>12</sup> The other algorithm, which relies on the method of images, is more closely related to radioisotropy than to beam tracing. It employs a priori elimination of reflection combinations that are not physically feasible and creates regions subject to the same reflection paths.<sup>8,10</sup> Steve Fortune introduced the use of triangulation to quickly determine which obstacles impede propagation, along with dynamic power culling of the rays, yielding a very fast algorithm.<sup>11</sup>

## References

1. V.O.K. Li and X. Qiu, "Personal Communication Systems," *Proc. IEEE*, Vol. 83, No. 9, Sept. 1995, pp. 1210–1243.
2. H. Hashemi, "The Indoor Radio Propagation Channel," *Proc. IEEE*, Vol. 81, No. 7, July 1993, pp. 943–968.
3. W. Honcharenko et al., "Mechanisms Governing UHF Propagation on Single Floors in Modern Office Buildings," *IEEE Trans. Vehicular Tech.*, Vol. 41, No. 4, Nov. 1992, pp. 496–504.
4. J.W. McKown and R.L. Hamilton, "Ray Tracing as a Design Tool for Radio Networks," *IEEE Network Magazine*, Vol. 5, Nov. 1991, pp. 27–30.
5. S.Y. Seidel and T.S. Rappaport, "Site-Specific Propagation Prediction for Wireless In-Building Personal Communication System Design," *IEEE Trans. Vehicular Technology*, Vol. 43, Nov. 1994, pp. 879–891.
6. R.A. Valenzuela, "A Ray Tracing Method to Predicting Indoor Wireless Transmission," *Proc. IEEE Conf. Vehicular Technology*, IEEE Press, Piscataway, N.J., 1993, pp. 214–218.
7. P. Ho and T.S. Rappaport, "Wireless Channel Prediction in a Modern Office Building Using an Image-Based Ray Tracing Method," *Proc. IEEE Globecom Conf.*, IEEE Press, 1993, pp. 1247–1251.
8. G.E. Athanasiadou, A.R. Nix, and J.P. McGeehan, "Indoor 3D Ray Tracing Predictions and Their Comparison with High Resolution Wideband Measurements," *Proc. IEEE Conf. Vehicular Technology*, IEEE Press, 1996, pp. 36–40.
9. S.J. Fortune et al., "WISE Design of Indoor Wireless Systems: Practical Computation and Optimization," *IEEE Computational Science & Engineering*, Vol. 2, No. 1, Spring 1995, pp. 58–68.
10. S. Takahashi et al., "An Evaluation Point Culling Algorithm for Radio Propagation Simulation Based on the Imaging Method," *Proc. Fifth Symp. Wireless Personal Comm.*, Virginia Tech., Blacksburg, Va., 1995, pp. 13.1–13.11.
11. S. Fortune, "A Beam-Tracing Algorithm for Prediction of Indoor Radio Propagation," *Applied Computational Geometry: Towards Geometric Engineering*, M. Lin and D. Manocha, eds., Springer-Verlag, New York, 1996, pp. 158–166.
12. A. Rajkumar et al., "Predicting RF Coverage in Large Environment Using Ray-Beam Tracing and Partitioning Tree Represented Geometry," *Wireless Networks*, Vol. 9, No. 2, June 1996, pp. 143–154.
13. H.D. Sherali, C.M. Pendyala, and T.S. Rappaport, "Optimal Location of Transmitters for Micro-Cellular Radio Communications System Design," *IEEE J. Selected Areas in Comm.*, Vol. 14, No. 4, May 1996, pp. 662–673.
14. A.R. Nix, G.E. Athanasiadou, and J.P. McGeehan, "Predicted Hiprelan Coverage and Outage Performance at 5.2 GHz and 17 GHz Using Indoor 3-D Ray Tracing Techniques," *Wireless Personal Comm.*, Vol. 3, No. 4, 1996, pp. 365–388.
15. P. Marinier, G.Y. Delisle, and L. Talbi, "Coverage Prediction Technique for Indoor Wireless Millimeter Waves System," *Wireless Personal Comm.*, Vol. 3, No. 3, 1996, pp. 257–271.
16. L. Talbi and G. Delisle, "FDTD Characterization of the Indoor Propagation," *J. Electromagnetic Waves and Applications*, Vol. 10, No. 2, 1996, pp. 243–247.
17. A. Lauer, A. Bahr, and I. Wolff, "FDTD Simulations of Indoor Propagation," *Proc. IEEE Conf. Vehicular Technology*, IEEE Press, 1994, pp. 883–886.
18. R.D. Murch et al., "A New Approach to Indoor Propagation Prediction," *Proc. IEEE Conf. Vehicular Technology*, IEEE Press, 1994, pp. 1737–1740.
19. J. Li, J.F. Wagen, and E. Lachat, "Preliminary Comparison of Two Propagation Prediction Methods for 2-D Microcells: Ray Tracing and TLM," *Proc. IEEE Conf. Vehicular Technology*, IEEE Press, 1996, pp. 31–35.
20. M.A. Panjwani, A.L. Abbot, and T.S. Rappaport, "Interactive Computation of Coverage Regions for Wireless Communication in Multifloored Indoor Environments," *IEEE J. Selected Areas in Comm.*, Vol. 14, No. 3, Apr. 1996, pp. 420–430.
21. Y. Qi et al., "Measurement and Simulation of Radio Wave Propagation in Two Indoor Environments," *Proc. IEEE Int'l Symp. Personal, Indoor, and Mobile Radio Comm.*, IEEE Press, 1995, pp. 1171–1174.

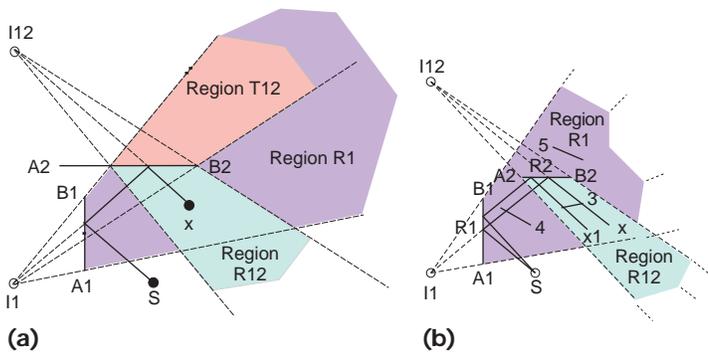


Figure 1. Two-reflection scenario: (a) reflection and transmission regions; (b) transmission with respect to a succession of images.

reflection path with the source. Figure 1a shows a two-reflection scenario. Given a source  $S$ , a first panel  $[A1, B1]$  with its associated image  $I1$ , and reflection region  $R1$ , we can create the image  $I12$  associated with a double reflection with another panel  $[A2, B2]$ . All points of region  $R1$  are illuminated by reflections of the source  $S$  from panel  $[A1, B1]$ . As a necessary condition for a double reflection, the reflection point on  $[A2, B2]$  must lie in region  $R1$ . If no such points exist, then panel  $[A1, B1]$  cannot see panel  $[A2, B2]$  in its region of reflection and therefore no higher-order reflections come from that panel. So, a point  $X$  must lie in a truncated triangular region  $R12$  determined by the two extremities of panel  $[A2, B2]$  seen in panel  $[A1, B1]$  and the image  $I12$ , to be illuminated by double reflections of  $S$  from panels  $[A1, B1]$  and  $[A2, B2]$ .

Given a set of  $N$  panels, we can precompute all single- and multiple-reflection regions. The problem is now one of inclusion: is a point  $X$  included in one or more of those regions? If the grid is uniform or has a regular structure, finding which receiver locations lie in a region is similar to scan-converting that region onto the grid.

### Transmission

Transmission (or refraction) refers to the effect of energy propagation by an electromagnetic wave through an obstacle. We must account for this energy-propagation mode because the direct line between a point  $X$  in a panel's reflection region and the actual reflection point  $R1$  on the panel might intersect other obstacles. For a transmission region to exist with respect to an image and a reflecting object, part of the transmitting panel must lie in the reflection region of the reflecting object. This condition is similar to the one required for an additional reflection with

respect to the same panel to exist. Therefore, when we compute the reflection regions of level  $L + 1$ , we can also compute the transmission regions of level  $L$ . Considering, for example, Figure 1a, the transmission region  $T12$  can be created at the same time as the reflection region  $R12$ . Any point in  $T12$  must necessarily be reached through panel  $[A2, B2]$ .

With a multiple-reflection scenario, having determined that a point  $X$  lies in the reflection region of a succession of panels, we must find which other panels are being traversed. We know that reflection points  $R1, \dots, RL$  with panels 1 to  $L$  exist, creating an  $L$ -reflection path between point  $X$  and source  $S$ , but we don't know the exact coordinates of those points. Let us consider a second-order reflection (see Figure 1b), where point  $X$  lies in the reflection region of image  $I12$ , which means that a path  $[X, R2], [R2, R1], [R1, S]$  must exist, although the coordinates of  $R1$  and  $R2$  are unknown. Points  $X$  and  $X1$  are arbitrarily close, although the corresponding paths to  $S$  do not intersect the same panels 3 and 4 and hence experience different transmission effects. Therefore, to compute transmission effects resulting from a point  $X$  and a succession of  $L$  reflections, we must compute the intersection points with the  $L$  reflecting panels and verify in which transmission regions they lie.

### Diffraction

In general, the diffraction mechanism radiates the energy in all directions, so its effects attenuate rapidly with distance. To include significant diffraction effects, we must consider only those regions where the diffracted energy cannot be neglected when compared with the energy of other modes such as transmission and reflection. Without line of sight and if the transmission effects through an obstacle are negligible because of higher reflectivity or absorption, diffraction effects will take over. The two propagation modes—diffraction and transmission—can illuminate a region behind an obstacle. As the energy due to transmission decreases, the diffraction phenomenon becomes more significant and must be considered.

As Figure 2 illustrates, a diffracting corner can be considered as a (possibly directive) source.<sup>2</sup> Given a transmitter location, a panel  $[A1, B1]$ , and its associated transmission region, we would want to determine grid points that are illuminated by a first-order diffraction from point  $A1$  or  $B1$ , and not covered by other more dominant propagation modes. Any point in panel  $[A1, B1]$ 's

transmission region can also be subject to diffraction by either of its edges. However, we must now account for transmission effects between the grid point and the diffracting corner. To do so, we must associate transmission regions with all panels in the transmission regions of points A1 or B1 as sources. Figure 2b shows the transmission region from point A1, and Figure 2a shows those associated with point B1. Now, for every grid point in region 1, we must find the transmitting panels between the grid point and the diffracting corner. Higher-order diffraction might be required if a panel also obstructs the line of sight between the grid point and the diffracting corner.

Because the addition of diffraction effects can become extremely cumbersome even if considering a few sources only, limiting their number is important. Such a procedure can be done during a preprocessing stage of the scene. The algorithm can then decide from which corners diffraction might be significant. Figure 3 shows the procedure for checking an L-reflection path. Given a reflection region R12 and a point X (stage 1 in Figure 3b), the algorithm checks whether X belongs in R12 (stage 2). If it does, it will reach the

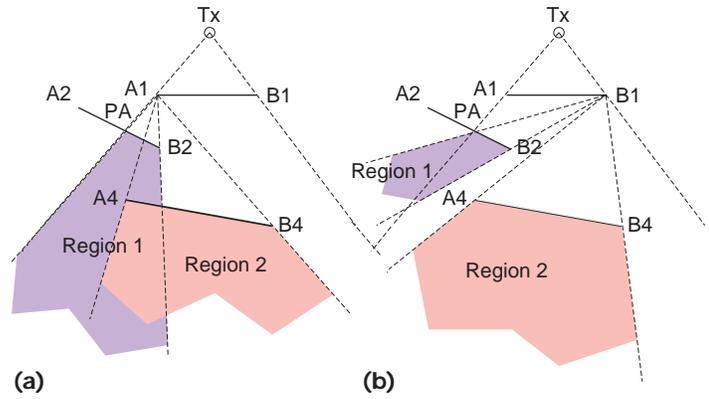


Figure 2. Regions associated with a significant diffraction (a) from corner A1 and (b) from corner B1.

source S via two reflections from panels P1 and P2. Now, we must verify which transmission or diffraction effects are associated with that double reflection (stage 3). If X belongs to region T12 or D12, transmission or diffraction effects will be generated from panel P1. The algorithm computes the intersection point with panel P2 (stage 4), continuing the process recursively until it reaches the source S (stages 5 and 6).

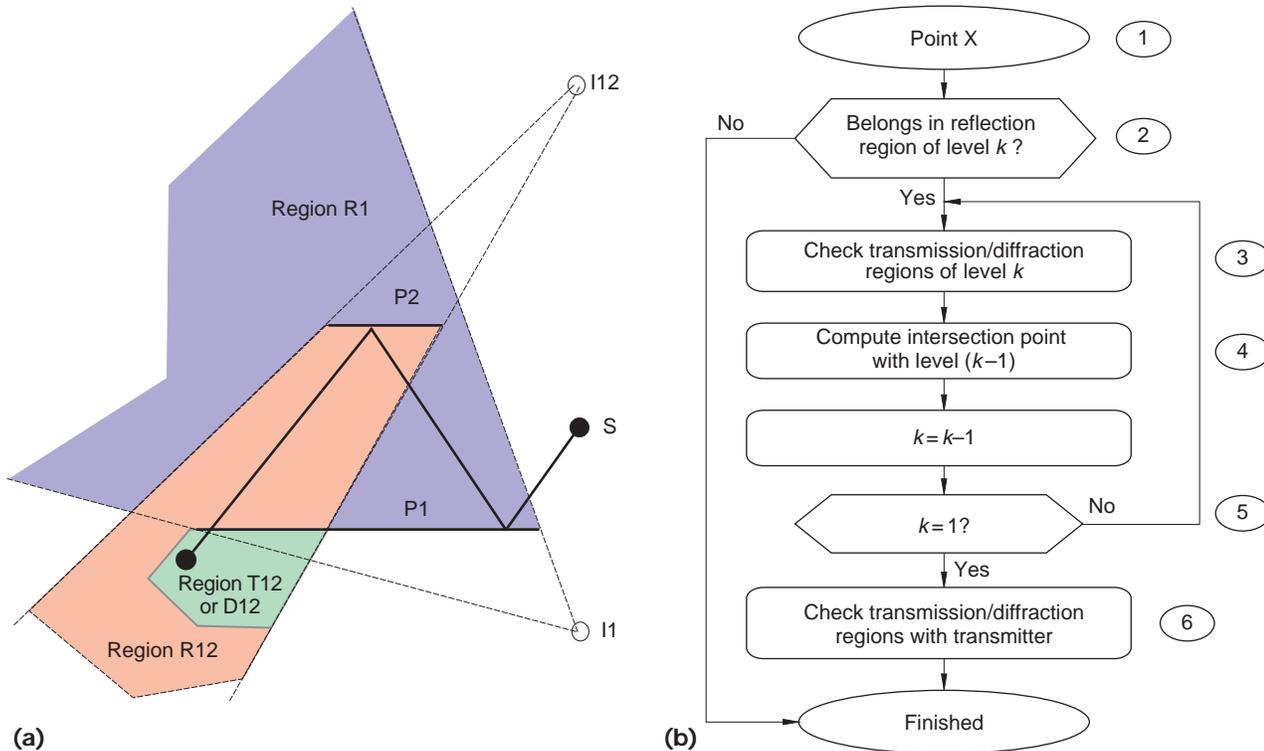


Figure 3. Procedure for checking an L-reflection path: (a) L-reflection path procedure; (b) flow chart of each L-reflection path procedure.

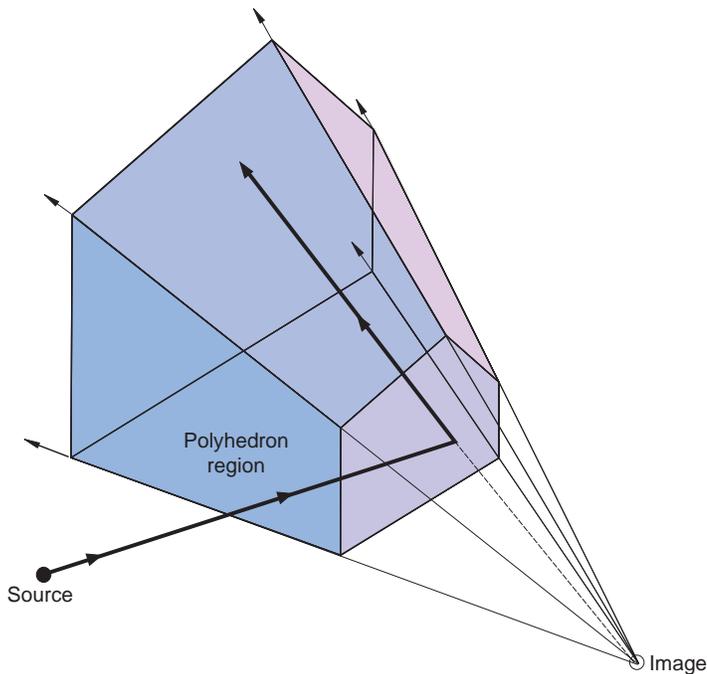


Figure 4. Three-dimensional reflection region.

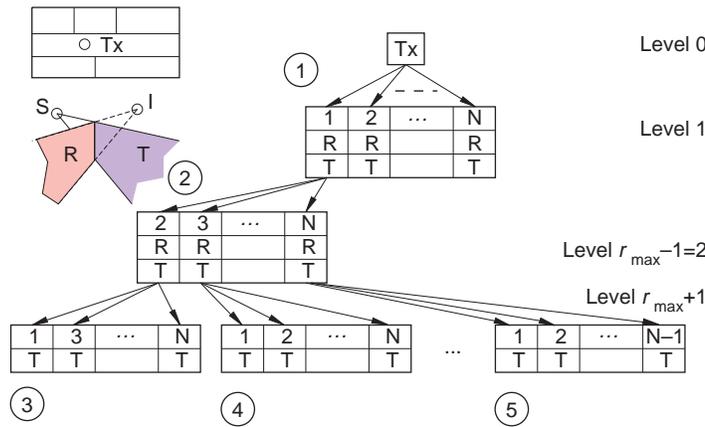


Figure 5. Data-flow graph of a tree branch.

### Generalization to three dimensions

Although the generalization of MR to 3D is conceptually straightforward, the computation involved is significantly more complex, demanding an efficient algorithm.

In 3D, each semi-infinite region will now transform into a semi-infinite polyhedra whose base is an  $N$ -sided polygon (see Figure 4). If we restrict the shape of the panels to be convex (a reasonable assumption because most objects considered are rectangular and any shape can be split into convex components), the part of a

panel an image sees through its associated panel consists of the intersection of an  $N$ -sided convex polygon with an  $M$ -sided convex semi-infinite polyhedra. Convexity is preserved and the result is a  $P$ -sided convex polygon. We can use clipping algorithms that take advantage of the convexity to determine those intersections efficiently.

The algorithm is truly 3D in that it evaluates the contribution of a 3D panel placed anywhere in the scene. It also considers multifloor scenarios or 3D building geometries, which is an advantage over algorithms that use, at some point, a 2D representation of the environment. Steve Fortune and his colleagues assume that each floor has a small number of cross sections (typically three) and use a 2D representation of each cross section via a triangulation.<sup>3</sup> Georgia Athanasiadou, Andrew Nix, and Joseph McGeehan assume single-floor scenarios and automatically add floor and ceiling reflections to the results of a 2D trace.<sup>4</sup> One type of architecture where those algorithms cannot extrapolate 3D from 2D representations without major modifications are split-level floors.

### Data-flow graph

The algorithm dynamically creates and eliminates *tree nodes*. A tree node is an image with its associated panel, region of reflection, transmission, and diffraction. To gain some insight into the MR algorithm's data flow, let's look at a *tree branch* (see Figure 5). The tree starts with the transmitter location.

1. The algorithm creates all first-level reflection and transmission regions.
2. It then uses the first node of the first level to create all the second reflections corresponding to that node. Because a panel cannot be reflected in itself, there will be at most  $N-1$  nodes—fewer if some panels are hidden from the view of the first level's first node.
3. Once again, the algorithm uses the second level's first node to create all third-level reflections and so forth. When considering a maximum number of reflections  $r_{max}$ , it will create all reflections of levels  $(0, \dots, r_{max})$  plus the associated transmission regions of level  $(1, \dots, r_{max} + 1)$ . To get the transmission regions of level  $L$ , the algorithm must process the tree's level  $L + 1$ . Referring to Figure 5 and letting  $r_{max} = 2$ , the algorithm creates nodes of sections 1, 2, and 3. Typically, at least two reflections are needed—three or four preferably.<sup>3</sup> Let  $n_{11}$  be node 1

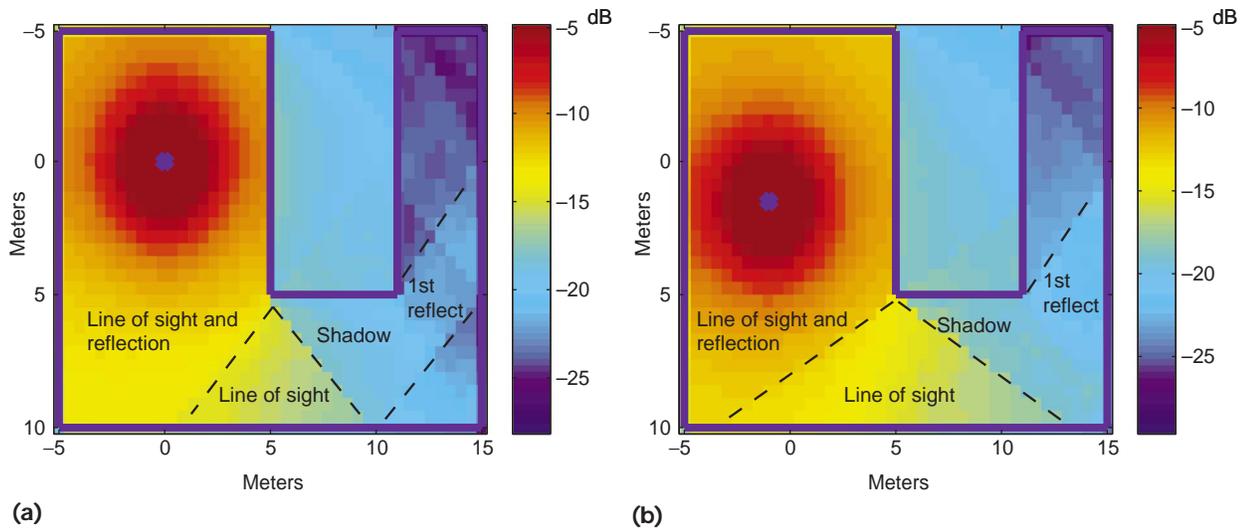


Figure 6. Sample coverage maps: (a)  $T_x = (0,0,1.5)$  and (b)  $T_x = (-1.0,1.5,1.5)$ . ( $T_x$  is transmitter location.)

of section 2. The algorithm will now examine node  $n_{11}$ , checking whether any grid point is included in its reflection region R. If yes, it will examine transmission. To check transmission of  $n_{11}$ , all nodes of sections 3, 2, and 1 are needed.

4. Once the algorithm has examined node  $n_{11}$ , all nodes of section 3 are deleted and node 2 of section 2  $n_{12}$  is used to create all nodes of section 4. Then it examines  $n_{12}$  with all grid points, deleting all nodes of section 4 and examining the next node of section 2. This process repeats until the algorithm reaches the last node of section 2 and has created, examined, and deleted all nodes of section 5. At that point, it examines node  $n_1$  with all transmission regions of section 2, deletes section 2, and restarts the process with node 2 of section 1,  $n_2$ .

### Sample results

Figure 6 illustrates two examples of attenuation coverage maps at 1.8 GHz in a very simple U-shaped room. The walls have a relative permeability  $\epsilon_r = 9.0$  (concrete). We chose the wall thickness to minimize transmission, resulting in a reflectivity, at frontal incidence, of 73% and a transmittivity of 27%. We assumed the walls to be smooth and lossless and considered up to four reflections with all associated transmission and diffraction effects. We neglected polarization and angle of incidence-related effects.

Four regions are apparent around the shadowing corner:

- the three regions of geometrical optics,
- the line of sight and reflection,
- the line of sight only, and
- the shadow region with respect to the corner.

We can also clearly see the first reflection off the bottom wall. Moving the transmitter slightly to the left and down by 1.8 m will create obvious changes, such as the line-of-sight region, but will also affect the first reflection in the branch of the U region. Even for a trivial scene, foreseeing the effect of a relatively small change of the transmitter location is not easy. Diffraction proved to be negligible because of the reflections off the bottom wall and transmission through the walls. Lowering the bottom wall's reflectivity did not increase the diffraction's importance. Only when we removed the bottom wall and made all other walls reflective only, was diffraction behind and close to the first corner significant. Any further diffracted effects would be below an indoor communication system's dynamic range.

### Gain and MI

The original MI creates a tree of all multiple-reflection permutations even if they are not physically realizable. In the process's second stage, the MI algorithm checks if a point creates a reflection with a panel and if that reflection exists. For a reflection to exist, the reflection point must lie on the panel.

We tested the MI and MR algorithms on two families of 3D scenes to evaluate their perfor-

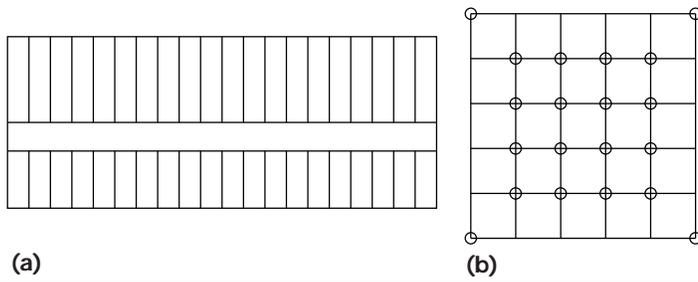


Figure 7. Two scenes for analysis: (a) rectangular and (b) square.

mance: a rectangular scene in which a corridor separates two rows of offices and a square scene comprising rooms in concentric squares more in the style of office towers (see Figure 7). We

increased the number of panels from 10 to 80 and 8 to 88, respectively, while considering up to three and four reflections. We laid a uniformly spaced 120-point grid over these scenes and constructed the region tree for each grid location and for, respectively, the 15 and six possible scene configurations. All quantities shown in the graphs are averages over these 120 grid points.

Figure 8 presents the acceleration gain defined as the ratio of the number of nodes to examine in the MI algorithm to that in the MR algorithm, as a function of the number of panels and reflections. These results suggest that MR will be between eight (for the rectangular scene) and 70 (for the square scene) times faster than MI (for

## Performance on a parallel machine

The total time to find all impulse responses consists of the sum of the time to build the tree and process all receiver locations. Speeding up the calculations of impulse responses for all receiver locations is crucial to obtain a result in real time.

All regions are independent of each other and can be treated in parallel. Rather than exploring the region tree serially, one node at a time, we could expand the tree as a linear array and check all nodes at the same time. Similarly, if we want to consider reflection and transmission effects, we must repeat the procedure to check an L-reflection path for all possible paths. This amounts to examining each node and its predecessors in the region tree one at a time. Once again, we could do all L-reflection path procedures in parallel. We could thus use a parallel machine to provide significant speedup gains, as researchers have done for outdoor propagation.<sup>1</sup>

### Speedup principles

A measure of an algorithm's parallelism is the *speedup factor*,<sup>2</sup> or the ratio of the time to execute processing on a single-processor machine  $T_1$  to the time processing takes on an  $n$ -processor machine  $T_n$ .

$$S = \frac{T_1}{T_n} \leq n$$

We want to achieve  $S = n$ . However, software issues such as the program's structure (namely, the irreducible serial part of a program) and hardware issues such as interprocessor communications and memory-access latency will lower the speedup factor. The art of parallel programming resides in getting  $S$  as close as possible to the maximum number of processors available on a system. This involves minimizing the serial-only portion of the code as well as masking inherent system latencies. As the number of processors grows from 1 to  $n$ , we can compute the corresponding actual speedup factor and obtain a

speedup curve. A highly scalable process will have a speedup curve close to the ideal  $S = n$  curve. Most of the processes, however, will exhibit some saturation point (Amdahl's law) beyond a certain value of  $n$ .

### Speedup on the Manna machine

We tested the MR algorithm on a distributed-shared-memory (DSM) MIMD machine called Manna, which has a multi-threaded architecture.<sup>3</sup> A *thread* is a sequence of instructions determined at compile time. Independent threads can run concurrently. Each node or processing element has a 50-MHz Intel i860 XP RISC processor with 32 Mbytes of dynamic RAM. Access to other nodes is through a bidirectional interface capable of handling 50 Mbytes/s in each direction for a total bandwidth of 100 Mbytes/s. A varying number of nodes can connect through  $16 \times 16$  crossbars. Manna comes with an automatic load balancer that automatically assigns threads to idle processors. The Threaded-C compiler uses a set of macros on top of a regular C program to track synchronization, control, and data locality.

The simplest approach assigns a first-level branch of the tree to all nodes. The system must read the scene and grid configuration, transmitter location, and maximum number of reflections and broadcast them to all nodes. It also computes and broadcasts the tree's first level to obtain the transmission regions associated with the transmitter. Then, it assigns a branch of the tree to each node until there are no more branches. Finally, it gathers the results from the various nodes and computes the coverage maps (attenuation and RMS delay spread).

Such a procedure will give ideal speedup curves only if the tree is evenly balanced. As Figure A shows, the coarse-grain speedup will scale well for the first nodes and then saturate to some value. The longest branches in the tree will govern the speedup factor. Increasing either the grid size or number of re-

three reflections and 80 panels), assuming that each node requires identical processing time. The difference in acceleration gains between both scenes results from the relative position of panels with respect to one another. The fewer panels that are visible on average from a given panel, the smaller the region tree.

Figure 9 shows that for three reflections and 80 panels, the rectangular scene has on average 24 transmission regions (that is, 24 panels on average visible from an image and its associated panel), which is almost four times the 6.3 transmission regions for the square scene. In a given scene family, the average number of transmission regions will increase with the number of panels but decrease as the number of reflections

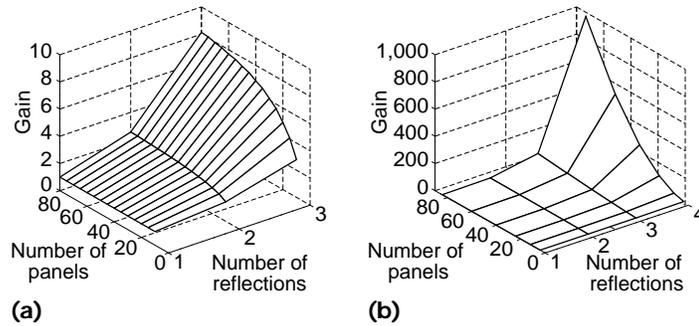


Figure 8. Acceleration gain expressed as the ratio of the number of nodes to examine in the MI to that in the MR: (a) rectangular scene and (b) square scene.

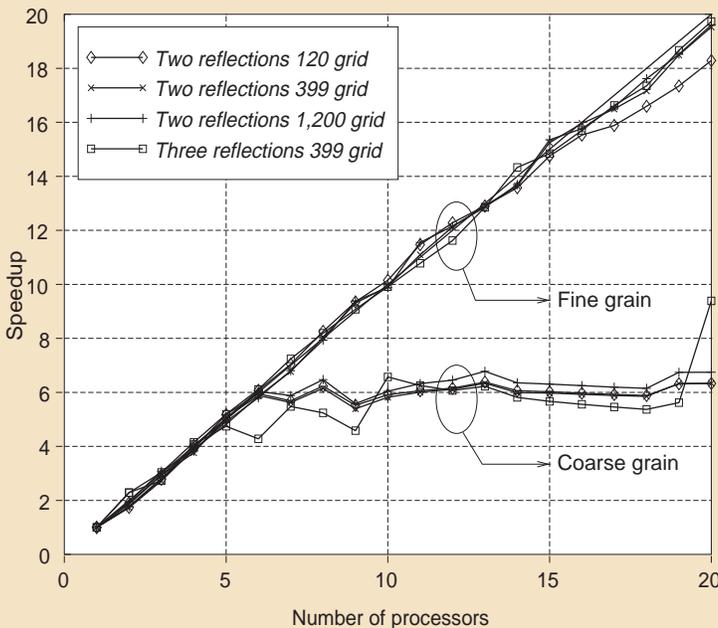


Figure A. Speedup curves for the Manna machine, by invoking one branch of the tree on each node (coarse grain) and subtrees on each node (fine grain).

flections will not improve performance unless a lucky assignment of the branch nodes occurs, as is the case for the last point of the three reflections curve, and results in an advantageous load balancing. We can still get linear speedup factors up to five. Because each branch of the tree is constructed independently, the only interprocessor communication is at the beginning of the process, during the broadcast of the scene and other parameters, and the end, when the grid results on each processor must be concatenated. So, we could use a

cluster of five or fewer workstations to create a parallel machine when the scene to process is too complex.

To remove such a dependency, we must reduce the computational grain size, making jobs smaller such that the computational load can be evenly distributed over all nodes, rather than a few selected ones. This will result in creating the subtree and processing each node of the subtree on a different machine node. We can use different strategies for assigning subtrees to nodes. We can rely on the automatic load-balancer, which will distribute the workload evenly among all nodes. Here, we obtain the fine-grain curves of Figure A. The superlinear speedup factors (that is, slightly better than ideal) for 9, 11, 12, and 15 processors result from memory-cache effects that deliver slightly different performance depending on how well the data fits in the cache. The more panels and larger the grid, the closer the speedup curves will be to ideal, which is exactly what we are looking for. The curves are close to the theoretical ones for three reflections. They are a bit worse for two reflections because of the additional overhead of two communications. Increasing the problem size will improve speedup factors.

#### References

1. P. Kuonen et al., "Parallel Computing of Radio Coverage," *Proc. IEEE Conf. Vehicular Technology*, IEEE Press, Piscataway, N.J., 1996, pp. 1438-1442.
2. K. Hwang, *Advanced Computer Architecture*, McGraw Hill, New York, 1993.
3. H.H.J. Hum et al., "A Study of the EARTH-MANNA Multithreaded System," *Int'l J. Parallel Programming*, Vol. 24, No. 4, Aug. 1996, pp. 319-347.

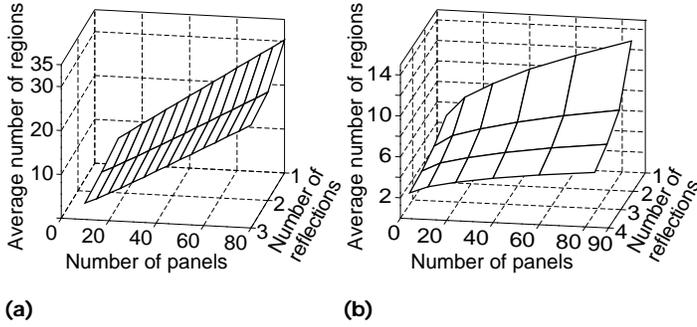


Figure 9. Average number of transmission regions for the two scene families from Figure 7: (a) rectangular scene and (b) square scene.

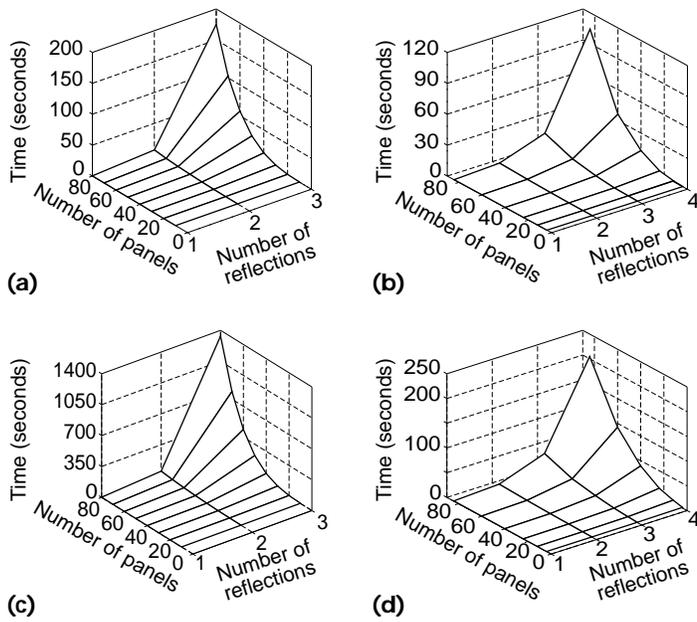


Figure 10. Time taken to build the region tree only (a and b) and process the scene (c and d).

increases. This is because the reflection regions are getting narrower on average for multiple reflections, and fewer panels are visible in those regions. Hence, the acceleration gains of MR with respect to the MI should increase as the number of reflections is increased.

An important parameter associated with MR is the visibility of the various panels. The more panels a potential reflector sees, the bigger the region tree and the bigger the number of transmission regions to examine. The visibility factor  $\overline{p_{s1}}$  is the average number of panels seen after one reflection. Table 1 shows that scenes with values

of  $\overline{p_{s1}}$  that are close require approximately the same processing time. Thus, the visibility factor can serve as a metric that indicates a scene's complexity.

With MR, we must go one level deeper to level  $L + 1$  to compute the transmission regions of level  $L$ . Therefore, the total size of the tree and storage needed can be more important than for MI. Consequently, building the tree will take longer for MR, especially because MR considers both reflection and transmission effects, while MI only considers reflection. The number of nodes at a level  $r_{max}$  in the image tree will be

$$N \prod_{j=1}^{r_{max}-1} (N-1),$$

where  $N$  is the number of panels in the scene. In the region tree, only those panels that can be seen from an image are left. Therefore, the number of nodes on average at a level  $r_{max}$  is

$$\overline{p_{s0}} \prod_{j=1}^{r_{max}-1} \overline{p_{sj}},$$

where  $\overline{p_{sj}}$  is the average number of panels seen after  $j$  reflections and  $\overline{p_{s0}} = N$ . The tree size ratio is then

$$\frac{\text{total \# nodes in region tree}}{\text{total \# nodes in image tree}} = \frac{\sum_{i=1}^{r_{max}} \overline{p_{s0}} \left( \prod_{j=1}^{i-1} \overline{p_{sj}} \right)}{\sum_{i=1}^{r_{max}} N \left( \prod_{j=1}^{i-1} (N-1) \right)}.$$

Because  $\overline{p_{s(j+1)}} \leq \overline{p_{sj}} < N$ , the number of nodes in MR does not grow as fast as in MI. Eventually, the ratio will fall below 1.

Figure 10 compares the time to create and process the tree for 120 grid points on a Sun/Sparc 10-41 workstation with a performance of approximately 50 Mflops/s (see the sidebar for a discussion of our algorithm's performance on a parallel machine). We found that MR is approximately 2.5 times faster than MI for the rectangular scene and 35 times faster for the square scene, with three reflections and 80 panels. As in all ray-shooting methods, we can reduce this time significantly by employing a dynamic power threshold rather than terminating the tree when a certain number of reflections has been reached. Furthermore, we can accelerate tree creation by using space-division techniques from computer graphics.

Table 1. Total time to process 120 impulse responses.

Scene	$N$	$p_{s1}$	Total time (seconds)		
			Two reflections	Three reflections	Four reflections
Square	44	9.62	1.51	8.33	39.37
Rectangular	20	9.56	1.47	9.77	35.91
Square	88	12.82	4.43	33.64	180.86
Rectangular	30	13	3.7	36.56	190.88

Significant research is invested around the world in speeding up the radio channel estimation process, in an effort to develop computer tools for optimizing the deployment of indoor wireless communication systems. Techniques from computational geometry and computer graphics could play a central role in such fast algorithms. Another promising direction for future consideration is the use of the MR algorithm (or its variations) on low-cost PC clusters, such as the Beowulf system developed by NASA. ❏

**Marc Kimpe** is with Adtran Inc., Huntsville, Alabama, working on systems for the High-Speed Digital Subscriber Loop. He received a BEng and a PhD from McGill University, both in electrical engineering. His PhD thesis considers algorithms for the estimation of indoor radio channels.

**Harry Leib** is an associate professor with the Department of Electrical and Computer Engineering at McGill University. His research interests include digital communications, wireless communications systems, and code division multiple access. He received his BSc and MSc from Technion, the Israel Institute of Technology, Haifa, Israel, and his PhD from the University of Toronto, all in electrical engineering. Contact him at the Dept. of Electrical Eng., McGill Univ., 3480 University St., Montreal H3A 2A7, Canada; leibh@wireless.ece.mcgill.ca.

## Acknowledgments

We thank the Computer Architecture and Parallel Systems Laboratory (CAPSL) team at the University of Delaware for letting us use the Manna machine.

**Olivier Maquelin** is a senior software engineer in Intel Corporation's Measurement, Architecture, and Planning Department. Earlier he worked as an associate researcher at McGill University, where he was one of the driving forces behind the Earth Multithreaded Architecture project. He received a PhD in electrical engineering at the Swiss Federal Institute of Technology.

## References

1. W.L. Stutzman and G.A. Thiele, *Antenna Theory and Design*, John Wiley & Sons, New York, 1981.
2. M.G. Sanchez et al., "Exhaustive Ray Tracing Algorithm for Microcellular Propagation Prediction Models," *Electronics Letters*, Vol. 32, No. 7, Mar. 1996, pp. 624–625.
3. S.J. Fortune et al., "WISE Design of Indoor Wireless Systems: Practical Computation and Optimization," *IEEE Computational Science & Engineering*, Vol. 2, No. 1, Spring 1995, pp. 58–68.
4. G.E. Athanasiadou, A.R. Nix, and J.P. McGeehan, "Indoor 3D Ray Tracing Predictions and Their Comparison with High Resolution Wideband Measurements," *IEEE Conf. Vehicular Technology*, IEEE Press, Piscataway, N.J., 1996, pp. 36–40.

**Ted H. Szymanski** is an associate professor at McGill University and a project leader at the Canadian Institute for Telecommunications Research, in Canada's Networks of Centers of Excellence program. His research interests include computing systems, optical networks, and telecommunications. He received a bachelor's degree in engineering science and his MASC and PhD in electrical engineering from the University of Toronto. He is a member of the IEEE Computer and Communications Societies.