This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2007 proceedings.

A Cube Oriented Ray Launching Algorithm for 3D Urban Field Strength Prediction

Rudolf Mathar, Michael Reyer, Michael Schmeink
Institute of Theoretical Information Technology
RWTH Aachen University
D-52056 Aachen, Germany
Email: {mathar,reyer,schmeink}@ti.rwth-aachen.de

***Abstract*— Fast radio wave propagation prediction is of tremendous interest for planning and optimization of cellular radio networks. We propose a cube oriented 3D ray launching algorithm for both fast and accurate field strength prediction, particularly suitable for urban scenarios. Our model allows field strength prediction for a 5 km^2 area with 5 meter resolution in about 8 sec with mean squared error of 7 dB. As urban environments cannot be completely described, we recommend a parameter calibration for different cities using measurement data from test runs.**

I. Introduction

Radio wave propagation models play an essential role in planning, analysis and optimization of radio networks. For instance, coverage analysis, interference estimation, channel and power allocation are based on field strength predictions, obtained from such models. These predictions are expected to be both accurate and fast, considering the vast amount of different configurations to select the best candidate from.

An overview of radio wave propagation models is given in [1] and [2]. Models proposed in the literature can be basically be divided into (semi) empirical and ray optical models. Semi empirical models calculate the received power on the basis of frequency, distance and an empirical part, mainly describing the obstacles' influence. Naturally, most models distinguish between prediction in and outside line of sight. The strength of such approaches is the speed of prediction. However, the prediction quality is low if the influence of deflection effects like diffraction, reflection and transmission is high. This leads to ray optical approaches, which identify ray paths through the scene to combat the lack of prediction quality at the cost of higher computation times.

In ray optical models the environment, e.g. buildings, is usually described by polyhedrons, formed of surface sections, called *facets* in the following. Several ray paths between the transmitter and receiver point are searched, regarding deflection effects as reflection on, transmission through and diffraction at edges of the given facets. Ray optical models are classified as ray tracing and ray launching, depending on the way the ray paths are determined.

In ray tracing models all possible ray paths from a receiver point to the transmitter are searched. The set of possible ray paths is limited by a maximum number of deflection points, i.e., points where deflection effects occur. For each receiver point the possible ray paths have to be recalculated, as there might be complete different ray paths. This leads to multiple calculation of nearly identical ray path pieces, particularly, if receiver points are nearby located. This results in precise predictions, but huge computational effort. Therefore, in [3] an extensive preprocessing is proposed which computes visibility of facets in advance. Hereby faster predictions are achieved.

Ray launching methods emit a finite set of rays from the transmitter in predetermined directions, cf. [4] and [5]. If rays hit a facet, possible deflection effects are performed. For diffraction it is necessary to emit a new ray bundle into the diffraction cone, whereas for reflection the direction has to be changed. A receiver point is hit if the ray path crosses its proximity. Each ray path is followed maximally once, but at the cost of precision. As the rays disperse, important deflection points or even receiver points may not be hit. Alternatively, in [6] and [7] 3D cones are used instead of single rays. Beyond this work, mixed models have been investigated, which follow partly rays and partly use empirical parameters, cf. [8]. Additional work on prediction algorithms, which is based on ray optical approaches, can be found in [9] and [10].

In the present paper we adopt a ray launching approach, as we are interested in field strength prediction in urban environments for a huge amount of receiver points. Our algorithm combines the precision of ray tracing methods with the speed of empirical algorithms, see also [11].

This paper is organized as follows. In Section II we introduce the functionality of the cube oriented ray launching algorithm. Our model for radio wave propagation is explained in Section III. Implementation details of our algorithm are given in Section IV. The propagation model depends on parameters which vary on different urban environments. Section V presents a parameter calibration method. Selected results are shown in Section VI. We compare quality and speed of field strength predictions in Section VII. Finally, Section VIII concludes this work.

II. Cube Oriented Ray Launching Algorithm

The main idea of our Cube Oriented Ray Launching Algorithm, CORLA for short, is to rasterize the given environment into cubes. A cube is called *filled* if it is intersected by a facet, see Section I. It is called a *horizontal or vertical diffraction source* if a facet boundary intersects with the cube. Additionally, subterrestrial cubes are called filled. Given two

1-4244-0353-7/07/$25.00 ©2007 IEEE

Algorithm 1 CORLA()

$\mathcal{C}$ ←GENERATEANDINITIALIZECUBES()
T ←GETTRANSMITTERDATA()
$\mathcal{R}$ ←GETRECEIVERPOINTS()
for all $r \in \mathcal{R}$ **do**
  $\mathcal{P}_r \leftarrow \emptyset$ {No receiver point is reached, yet.}
end for
$p \leftarrow T$.GETPOSITION() {Initializes ray path p.}
EVALCUBES(p,$\mathcal{C}$,0)
{$\mathcal{P}$ contains sets of possible ray paths for each receiver point.}

points p_1 and p_2, p_2 is called *visible to* p_1 if no cube on the straight line between p_1 and p_2, other than the cubes containing either p_1 or p_2, is filled.

An abstract program flow is given in Algorithms 1 and 2. In Algorithm 1 all cubes are generated, initialized and stored in $\mathcal{C}$. The transmitter data T and the set of receiver points $\mathcal{R}$ are loaded. For each receiver point r the set of paths leading to r, named $\mathcal{P}_r$, is initialized with the empty set. Paths are described by sequences of points beginning with the transmitter and ending with the *current point*. Intermediate points represent sources of deflection. The algorithm EVALCUBES is invoked with the currently active path p, the set of cubes in the deflection cone $\mathcal{C}_A$ and recursion depth d. All candidates from deflection cone $\mathcal{C}_A$ are checked up on visibility to the current point of the path. For all visible points r the set $\mathcal{P}_r$ is updated. If the current recursion depth is less than a maximum d_{max}, EVALCUBES is called for all deflection cones with an updated ray path and incremented recursion depth. Finally, Algorithm 1 provides all ray paths $\mathcal{P} = \{\mathcal{P}_r \mid r \in \mathcal{R}\}$.

III. Model for Radio wave propagation

The attenuation in line of sight of a receiver point r with distance $d(r)$ is given by the ratio between transmission power P_t and the received power $P_0(r)$.

$$\begin{aligned} L_0(r) &= \frac{P_t}{P_0(r)} = \frac{\hat{P}_t (4\pi)^2 d(r)^\gamma}{\lambda_t^2 G_t(\phi(r),\psi(r))}, \\ L_0^{\text{dB}}(r) &= 20 \lg \frac{4\pi}{\lambda_t} - 10 \lg G_t(\phi(r),\psi(r)) \\ &\quad + z_A + 10\gamma \lg d(r), \end{aligned}$$

where λ_t denotes the signal wavelength, $G_t(\phi(r),\psi(r))$ the antenna gain in direction $(\phi(r),\psi(r))$ and γ is the path loss exponent. Note that the precise antenna power P_t is usually unknown. Hence, our model includes estimators $\hat{P}_t$ and z_A, respectively. The attenuation $L(r)$ at a receiver point $r \in \mathcal{R}$ is given by

$$L(r) = \sum_{p \in \mathcal{P}_r} L(p). \tag{1}$$

The attenuation $L(p)$ has to be evaluated with the *Uniform Geometrical Theory of Diffraction*, cf. [12]. Since parameters, like the type of surface or material, are usually not available simplified attenuation functions considering merely the overall

Algorithm 2 EVALCUBES(RayPath p, CubeSet $\mathcal{C}_A$, Recursion-Depth d)

$t \leftarrow p$.GETCURRENTPOINT()
for all $c \in \mathcal{C}_A$ **do**
  if c.ISVISIBLETO(t) **then**
    $\hat{p} \leftarrow$p + c {Add visible c at end of ray path.}
    {Add newly found paths to receiver points.}
    for all $r \in \mathcal{R}$ **do**
      if r.ISINCUBE(c) **then**
        $\mathcal{P}_r$.ADDPATH($\hat{p}$)
      end if
    end for
    if c.ISFILLED() AND ($d < d_{max}$) **then**
      $\mathcal{C}_T$ ←GETCUBESINTRANSMISSIONCONE($\hat{p}, \mathcal{C}_A$)
      EVALCUBES($\hat{p}, \mathcal{C}_T, d+1$)
      $\mathcal{C}_R$ ←GETCUBESINREFLECTIONCONE($\hat{p}, \mathcal{C}_A$)
      EVALCUBES($\hat{p}, \mathcal{C}_R, d+1$)
      if c.ISVERTICALDIFFRACTIONSOURCE() **then**
        $\mathcal{C}_V$ ←GETCUBESINVERTICALCONE($\hat{p}, \mathcal{C}_A$)
        EVALCUBES($\hat{p}, \mathcal{C}_V, d+1$)
      end if
      if c.ISHORIZONTALDIFFRACTIONSOURCE() **then**
        $\mathcal{C}_H$ ←GETCUBESINHORIZONTALCONE($\hat{p}, \mathcal{C}_A$)
        EVALCUBES($\hat{p}, \mathcal{C}_H, d+1$)
      end if
    end if
  end if
end for

path length and the direction changes at deflection points are used, cf. [13].

In our model we distinguish between different deflection effects. Given a deflection point and its direction change α, the attenuations due to reflection, vertical diffraction, horizontal diffraction and transmission are given by

$$\begin{aligned} L_R^{\text{dB}}(\alpha) &= \sum_{j=0}^{k} z_{R,j}\ \alpha^j, \\ L_V^{\text{dB}}(\alpha) &= \sum_{j=0}^{k} z_{V,j}\ \alpha^j, \\ L_H^{\text{dB}}(\alpha) &= \sum_{j=0}^{k} z_{H,j}\ \alpha^j \text{ and} \\ L_T^{\text{dB}}(\alpha) &= \sum_{j=0}^{k} z_{T,j}\ \alpha^j \overset{\alpha=0}{=} z_{T,0} = z_T, \end{aligned}$$

where k denotes the maximal degree of the polynomials. Since the refraction index is usually unknown, it is neglected by choosing $\alpha = 0$. Note that the types of deflection and the direction changes may be either identified given a ray path or stored while creating the ray paths. The number of deflection points for each deflection effect on a path p are given by $n_R(p), n_V(p), n_H(p)$ and $n_T(p)$. In summary, the attenuation

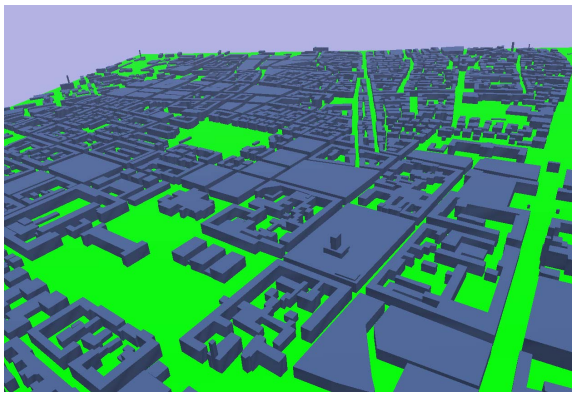


Fig. 1. 2.5 dimensional building data of Munich

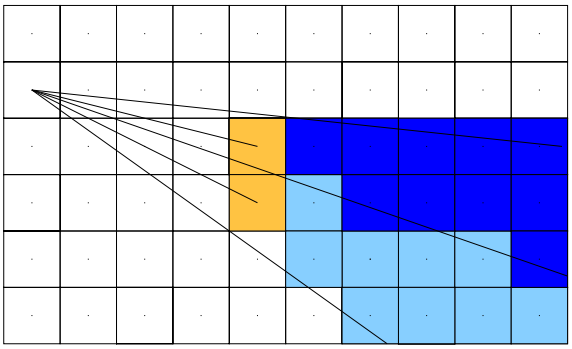


Fig. 2. Shadow of two neighboring cubes

of a ray path p is given by

$$L^{\text{dB}}(p) = L_0^{\text{dB}}(r) + n_T(p)z_T + \sum_{i=1}^{n_R(p)} L_R^{\text{dB}}(\alpha_{R,i}(p)) + \sum_{i=1}^{n_V(p)} L_V^{\text{dB}}(\alpha_{V,i}(p)) + \sum_{i=1}^{n_H(p)} L_H^{\text{dB}}(\alpha_{H,i}(p)). \quad (2)$$

The search for ray paths is simplified and consequently accelerated if (1) is approximated using the *strongest ray path* with minimal attenuation,

$$L(r) \approx L_{\min}(r) = \min_{p \in \mathcal{P}_r} L(p). \quad (3)$$

According to the following, the approximation error of (3) is low. Considering two equally strong ray paths, the attenuation would decrease by 3 dB when correctly added. If one of the ray paths is just 25% longer, the attenuation decrease would be at about 1.2 dB. Three more 25% longer paths would still obtain an attenuation decrease less than 3 dB. This approximation seems to be reasonable regarding that in [1] predictions with mean squared error (MSE) of 5 to 10 dB are considered as excellent. However, such effects are mitigated by carefully chosen model parameters.

IV. CORLA Implementation

The C++ implementation of CORLA uses the radio wave propagation model of the last section. It has been optimized with regard to speed but keeps accuracy at a high level, cf. Sections VI and VII. This has been achieved by exploiting the cube oriented data structure, by avoiding redundant computations and by skipping rare events. This section gives insight to some of the underlying principles.

Typically, input data is given by building data in a 2.5 dimensional format, i.e., each building is described by its polygonal outline and a single height, see Fig. 1. Additionally, ground level information is considered. Given such data the cube information has to be created.

In order to speed up the CORLA implementation, exact geometrical data is replaced as much as possible by approximations from the cubical structure. Instead, we use a discrete shadowing principle as depicted in Fig. 2. A cube is marked as *shadowed* whenever its center lies in the shadow cone of some filled cube. Double marking is avoided by using the bisecting line between the centers of neighboring cubes. Using cube edges instead of the facet borders leads to errors in the shadowing process. However, these are usually negligible since only small diffraction angles and according attenuations are affected.

Observations demonstrate that for most receiver points in an urban scenario a nearly strongest path exists with no joint horizontal and vertical diffraction. Hence, computational speed is enhanced by disregarding changes from horizontal to vertical diffraction or vice versa, i.e., for all paths p either $n_V(p) = 0$ or $n_H(p) = 0$ holds, cf. (2).

Assuming a certain attenuation bound for relevant ray paths and that the attenuation is monotone increasing along ray paths, a lot of rays may be dropped at early stages before reaching the recursion depth limit.

V. Parameter Calibration

Due to limited environmental information, ray optical models do not describe influences by building materials, vegetation, style of roofs, et al. A city with modern skyscrapers, mainly with glass fronts and flat roofs, and a small town, mainly with pitched roofs and stone fronts, will show different attenuation patterns, which, however, can be represented by the right choice of a parameter set in the above model. We now introduce a parameter calibration procedure for finding suitable parameters for each type of city. Additionally, the parameter calibration implicitly compensates the approximation for attenuation in (3) and the simplifications described in Section IV.

An introduction to multivariate analysis and least squares estimation used for calibration is given in [14]. All parameters introduced in Section III are merged into a vector $\mathbf{z} = (z_A, \gamma, z_T, z_{R,0}, \ldots, z_{R,k}, z_{V,0}, \ldots, z_{V,k}, z_{H,0}, \ldots, z_{H,k})$. By physical reasons, the set of parameter vectors has to be restricted, e.g., γ is assumed to be in the interval $[2, \ldots, 5]$. $\mathcal{Z}$ denotes the set of all feasible parameter vectors. Further, a set $\mathcal{M} \subset \mathcal{R}$ of measurement points with $M := |\mathcal{M}|$ and attenuation measurements $L_{\mathcal{M}}^{\text{dB}}(m)$ are given. The attenuation dependences on this vector are indicated by writing $L(r, \mathbf{z})$

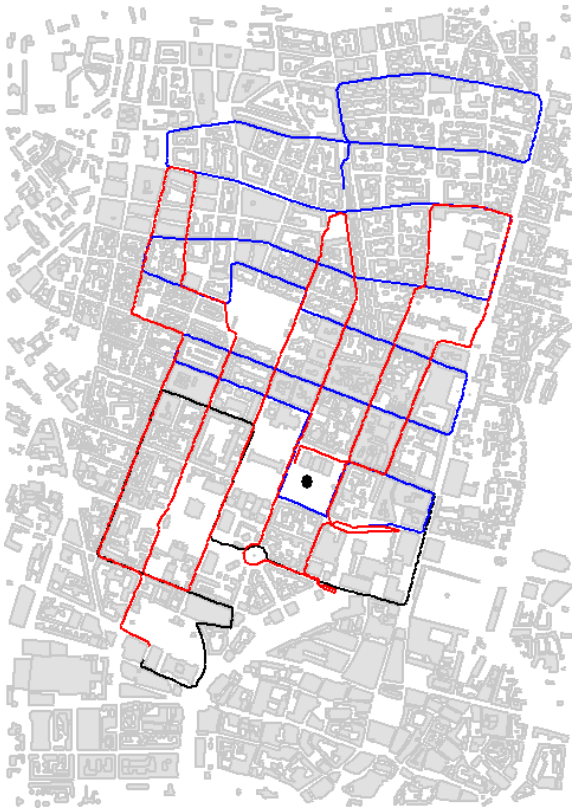


Fig. 3. Test runs 1, 2 and 3 in Munich, transmitter at black dot

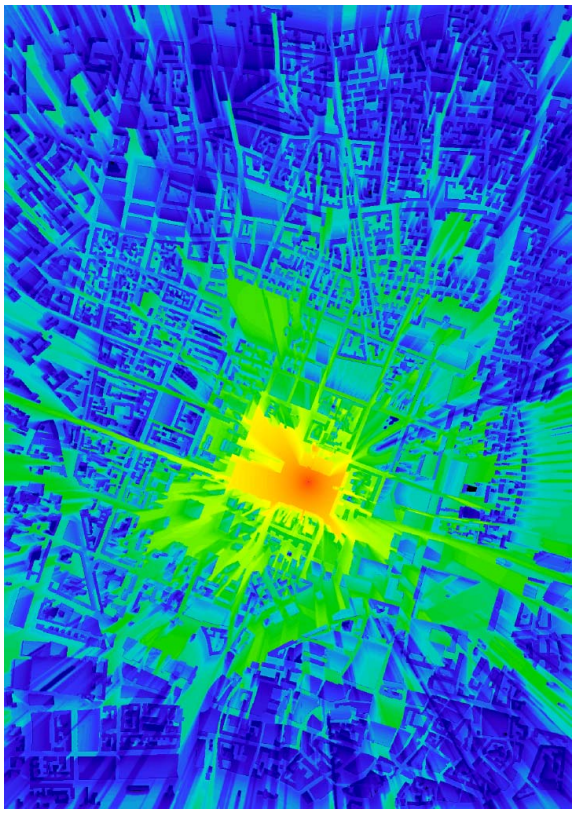


Fig. 4. Field strength prediction in Munich

and $L(p, \mathbf{z})$. Commonly, the mean squared error

$$q(\mathcal{M}, \mathbf{z}) := \sqrt{\frac{1}{M} \sum_{r \in \mathcal{M}} \left[L^{\text{dB}}_{\mathcal{M}}(r) - L^{\text{dB}}(r, \mathbf{z})\right]^2}$$

is used as a performance index. It is minimized by

$$\mathbf{z}^* := \arg\min_{\mathbf{z} \in \mathcal{Z}} \sum_{r \in \mathcal{M}} \left[L^{\text{dB}}_{\mathcal{M}}(r) - L^{\text{dB}}(r, \mathbf{z})\right]^2 .$$

It is easy to see that $L^{\text{dB}}_{\text{min}}(r, \mathbf{z})$ is locally linear in $\mathbf{z}$. In fact, it is linear in $\mathbf{z}$ as long as changes in $\mathbf{z}$ do not lead to a change of the strongest ray path. Taking this into account, we cyclically iterate between parameter estimation and computation of strongest ray paths. A strict proof of convergence is difficult to achieve for this alternating approach. However, in practice two to three iterations lead to good results as is shown in the next section.

This model is enhanced by replacing the path loss exponent γ with path loss exponents z_L and z_N. We use z_L if the receiver points lie in line of sight and z_N otherwise.

VI. Results

In the COST-231 project, cf. [1], measurements have been conducted in the city of Munich. The building data and measurement from three test run are available, cf. [15] and Figures 1 and 3. All test runs have been realized with an omnidirectional antenna at a single transmitter position. The parameter calibration has been performed using the third test run. These parameter have been applied to evaluate CORLA with the first and second test run. We assumed linear attenuation due to the direction changes ($k = 1$), as quadratic functions did not lead to significant improvements. As start configuration we used the parameter vector $\mathbf{z} = (0, 2, 2, 15, 6, 0, 6, 0, 6, 0)$. The path loss exponents are initialized with two. The loss due to transmission is 15 dB and for each other deflection there is an angle independent loss of 6 dB. Using those parameters, the mean squared error is about 8 dB. For numerical optimization using OPT++, cf. [16], the parameters are restricted by absolute values 20 and 5 for z_A, respectively. After two iterations we achieved the stable parameter vector $\mathbf{z}^* = (-5, 2.76, 3.06, 15, 16, 1.2, 5.07, 0.12, 2, 0.15)$. Note that the transmission loss has not changed, as no strongest ray path contained a transmission effect. The attenuation effect due to reflection is striking by high. Further investigation of the COST-scenario revealed reflection effects to be rare, such that the parameter calibration does not produce reliable parameters for that effect. The mean squared error for test run three improved to 5.7 dB. The resulting radio wave propagation for Munich city is shown in Fig. 4. How precision and computation time affect each other for different cube size resolutions is shown in Table I. Note that the parameter calibration is carried out only once for each type of city. Parameter calibration needs at most three predictions by CORLA with according running time. The parameter estimation time itself is negligible.

CORLA / MSE	Resolution 5 m	Resolution 2.5 m
Runtime (P4 1.8 GHz)	<10 s	<30 s
MSE of test run 1	7.0 dB	6.1 dB
MSE of test run 2	4.5 dB	4.2 dB
MSE of test run 3*	6.1 dB	5.7 dB
Mean value for test runs 1 and 2	5.8 dB	5.2 dB

* = Calibration with this test run

TABLE I
CORLA QUALITY AND SPEED FOR DIFFERENT RESOLUTIONS

Algorithm / MSE	Test runs 1 [dB]	Test runs 2 [dB]	Mean of test runs 1 and 2 [dB]
Ericsson (RT+WI)	6.7	7.1	6.9
France Telecom (SE)	6.9	9.5	8.2
Swiss Telecom (RT)	14.6	15.5	15.0
COST (WI)	7.7	**5.9**	6.8
Univ. Valencia (WI)	8.7	7.0	7.8
Telecom Italia (SE)	10.4	12.3	11.3
Swiss Telecom (SE)	7.0	6.2	**6.6**
Univ. Bologna (RT)	**6.3**	10.9	8.6
Univ. Karlsruhe (RT)	8.5	9.1	8.8
CORLA	**6.1**	**4.2**	**5.2**

RT = ray tracing, WI = Walfisch-Ikegami, SE = semi empirical

TABLE II
QUALITY COMPARISON COST-MUNICH

VII. COMPARISONS

There are two magnitudes of interest to benchmark radio wave propagation models, quality and speed. In [1] quality is compared by the mean squared error. Results from the COST-project are compared to our results in Table II. The COST-report includes both ray optical and semi empirical algorithms. For both test runs CORLA outperforms its competitors. It is obvious that COST-participants with good results for one test run achieve only mediocre results for the other test run. Therefore, in the last column the mean of both test runs is shown. CORLA is clearly superior to the other algorithms with respect to this criterion.

Algorithm	Resolution [m]	MSE [dB]	Preprocessing [sec]	Runtime [sec]
WinProp, UDP	10	7.99	several*	36*
WaveSight	5	6.2	—	297**
CORLA	5	6.1	—	8***

* = on AMD™Athlon™2800+
** = on Pentium™3 600 MHz, 4 km^2
*** = on Pentium™4 1800 MHz

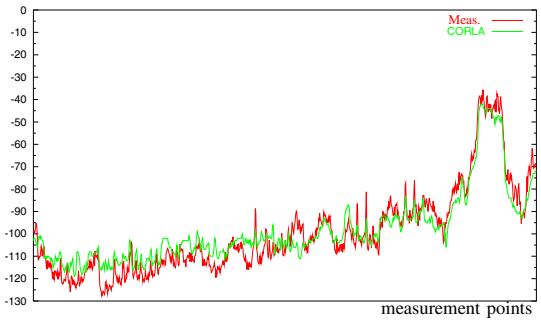
TABLE III
RUNTIME COMPARISON FOR TEST RUN 3 IN MUNICH (5 km^2)

The algorithms' mean squared errors may not be arbitrarily reduced. On one hand the environmental data is incomplete, like the style of roofs and the type of surface or building material. Further, there are random effects which cannot be covered by a manageable mathematical model at all. On the other hand measurements themselves are imprecise. Two consecutive test runs within cities under comparable conditions exhibit a mean squared error deviation of over 3 dB, cf. [17], chapter 3.2.3.3. Having all those effects in mind, algorithm CORLA reveals excellent performance.

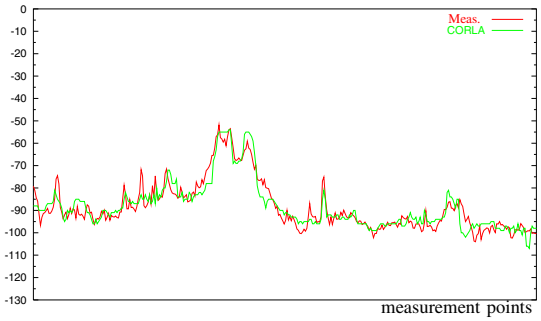
The comparison between CORLA predictions and measurements for test runs 1, 2 and 3 are shown in Fig. 5. Most of the effects observed in the measurements are approximately predicted by the algorithm CORLA.

We have further compared CORLA with two commercial solutions announcing extreme fast running times. Both solutions refer to the above COST-scenario, such that a direct comparison with CORLA is feasible. The results are comprised in Table III. Computation times have been observed on different processor architectures such that a comparison of times has to be taken with care. Nevertheless, CORLA seems to achieve higher accuracy at shorter running times than its competitors, which are shortly introduced in the following paragraph. We assume a well calibrated set of model parameters for the type of city Munich belongs to. Therefore the time needed for parameter calibration is not taken into account.

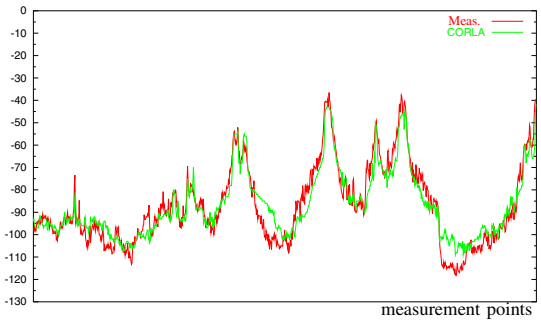
WinProp, cf. [18], provides multiple algorithms with different accuracies and speeds. These algorithms include COST-Walfisch-Ikegami, Urban Dominant Path (UDP) and Intelligent Ray Tracing (IRT). The IRT algorithm starts with a preprocessing dividing the facets into tiles and segments, calculating visibility between those tiles and segments and storing them into a database, cf. [9]. The exhaustive and redundant search for visible facets is avoided at runtime by using this database, however, at the cost of large computation times for the one-time preprocessing. It takes several hours up to a day depending on the scenario and resolution. As algorithm UDP achieves better results with respect to the mean squared error and is comparably fast to algorithm IRT, only this version is taken into account.



(a) Test run 1: Mean error 1.0 dB, MSE 6.1 dB



(b) Test run 2: Mean error 0.1 dB, MSE 4.2 dB



(c) Test run 3: Mean error 0.1 dB, MSE 5.7 dB

Fig. 5. Predictions versus measurements for test runs 1, 2 and 3 according to the tracks shown in Fig. 3

The company WaveCall uses a ray tracing method. Running times are downsized by a smart reduction of candidates for being visible objects, based on selecting important ray paths, cf. [19]. The accuracy and running time of the prediction tool WaveSight are shown in Table III, cf. [20].

VIII. Conclusions

In this paper, we have presented the cube oriented ray launching algorithm CORLA, a fast tool for field strength prediction. It offers the precision of ray optical methods at running times comparable to empirical predictions. Our model allows field strength prediction for a 5 km^2 area with 5 meter resolution in about 8 sec with mean squared error of 7 dB in the COST-Munich scenario.

The full potential of our approach, however, is not yet exploited by the current implementation. Future research will be devoted to employing cheap but extremely powerful hardware as is presently available in multi core processor architectures and graphical processor units. Furthermore, deep parallelization on this hardware will significantly speed up computation times. Network operators often avoid standard ray optical methods because of their long running times. CORLA contributes to overcome this problem.

References

[1] E. Damosso, Ed., *COST Action 231: Digital mobile radio towards future generation systems, Final Report*. Luxembourg: Office for Official Publications of the European Communities, 1999.

[2] N. Geng and W. Wiesbeck, *Planungsmethoden für die Mobilkommunikation*. Springer, 1998.

[3] G. Wölfle, R. Hoppe, and F. Landstorfer, "A fast and enhanced ray optical propagation model for indoor and urban scenarios, based on an intelligent preprocessing of the database," in *Proceedings PIMRC*, Osaka, Japan, 1999.

[4] G. Durgin, N. Patwari, and T. S. Rappaport, "An advanced 3D ray launching method for wireless propagation prediction," in *Proceedings IEEE VTC Spring*, Phoenix, AZ, 1997, pp. 785 – 789.

[5] M. Schmeink and R. Mathar, "Preprocessed indirect 3D-ray launching for urban microcell field strength prediction," in *Proceedings IEEE AP*, Davos, Switzerland, 2000.

[6] M. Nidd, S. Mann, and J. Black, "Using ray tracing for site-specific indoor radio signal strength analysis," in *Proceedings IEEE VTC Spring*, Phoenix, AZ, 1997, pp. 795–799.

[7] T. Frach, "Adaptives hierarchisches Ray Tracing Verfahren zur parallelen Berechnung der Wellenausbreitung in Funknetzen," Ph.D. dissertation, RWTH Aachen, 2003.

[8] J. Beyer, "Ausbreitungsmodelle und rechenzeiteffiziente Methoden für die Feldstärkeprognose in städtischen Mikrozellen," Ph.D. dissertation, Universität-Gesamthochschule Siegen, 1997.

[9] R. Wahl, G. Wölfle, P. Wertz, P. Wildbolz, and F. Landstorfer., "Dominant path prediction model for urban scenarios," in *14th IST Mobile and Wireless Communications Summit*, 2005.

[10] P. Wertz, R. Wahl, G. Wölfle, P. Wildbolz, and F. Landstorfer., "Dominant path prediction model for indoor scenarios," in *German Microwave Conference (GeMiC)*, 2005.

[11] M. Schmeink, "Optimierungsalgorithmen zur automatisierten Funknetzplanung." Ph.D. dissertation, RWTH Aachen University, 2005.

[12] T. Kürner, D. J. Cichon, and W. Wiesbeck, "Concepts and results for 3D digital terrain-based wave propagation models: an overview," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 7, pp. 1002–1012, 1993.

[13] G. Wölfle, R. Hoppe, T. Binzer, and F. M. Landstorfer, "Radio network planning and propagation models for urban and indoor wireless communication networks," in *Proceedings IEEE AP*, Davos, Switzerland, 2000.

[14] K. V. Mardia, J. T. Kent, and J. M. Bibby, *Multivariate Analysis*. London: Academic Press, 1979.

[15] "Cost 231. Urban micro cell measurements and building data," http://www.ihe.uni-karlsruhe.de/forschung/cost231/cost231.en.html.

[16] "OPT++. Nonlinear Optimization Library," Sandia National Laboratories, USA, http://csmr.ca.sandia.gov/opt++/.

[17] K. Rizk, "Propagation in microcellular and small cell urban environment," Ph.D. dissertation, Ecole Polytechnique de Lausanne, 1997.

[18] AWE Communications GmbH, Stuttgart, "WinProp," http://www.awe-communications.com.

[19] K. Rizk, R. Valenzuela, S. Fortune, D. Chizhik, and F. Gardiol, "Lateral, full-3D and vertical plane propagation in microcells and small cells," *IEEE Transactions on Vehicular Technology*, vol. 48, no. 2, pp. 998–1003, 1998.

[20] "WaveSight," WaveCall SA. http://www.wavecall.com.