Analytical Derivation of the Reduction of Computation Time by the Voxel Crossing Technique

Uwe Martin Stephenson
University for Applied Sciences, D 22297 Hamburg, Germany, email: U.M.Stephenson@t-online.de

Introduction

The practical application of room acoustical simulation techniques as ray tracing still suffers from high computation times. Different algorithms to reduce the time-consuming search for ray–wall-intersections have been tested, mainly basing on spatial subdivision as known from computer graphics. For room acoustics, where the number K_0 of surface polygons (walls) is not so high, the voxel technique appears suitable. Allthough this technique is known from computer graphics [5] and was published also in the field of room acoustics [1-4] its reduction of computation time was not yet investigated up to now (by the author derived but presented only orally in 1990 [2]).

Method

By the voxel technique (opposite to hierarchical methods as for ex. the octree method [5]) the space or better a surrounding box of volume V, fig.1. is subdivided *regularly* in a grid of small cubic volume elements (voxels) of size d.

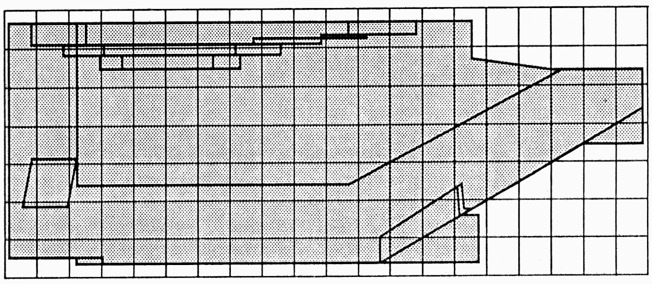

Figure 1: example of a room quantized into voxels

So, a "degree of spatial quantization" n_q may be defined by

$$n_q = \sqrt[3]{V} / d \tag{1}$$

Other needed parameters beside the room surface S are the mean free path length $\Lambda = 4V/S$ (2)

A formfactor is $\sigma = \sqrt[3]{36\pi} \bullet V^{2/3}/S \approx 4.836 V^{2/3}/S$ (3)

such that $\Lambda \approx 0.827 \bullet \sigma \bullet \sqrt[3]{V}$ (4)

For simplicity, for rooms of about proportion 1:2:4, in the following it is assumed $\sigma \approx 0.7$ (for a sphere $\sigma = 1$).

The idea: Only if a voxel intersects a wall, the intersection point needs to be computed. With high n_q, the advantage is obvious: most voxels are "empty", i.e. not intersected by walls.

These informations (how many and which walls intersect each voxel) have to be pre-computed. This is once performed by scanning each surface of the room by a set of parallel lines built by its intersection with the many parallel surfaces of the 3-dimensional voxel grid. These lines serve as rays such that scanning is simply like ray tracing (fig.2.) but with a storing effect.

During the actual ray tracing, the voxels are crossed step by step. This "voxel crossing algorithm" (fig.2), was

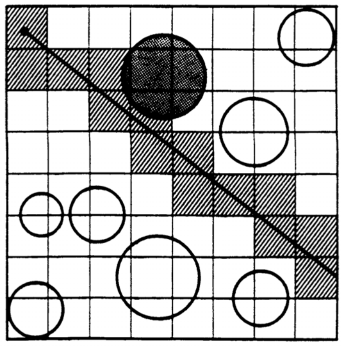

Figure 2: voxel crossing by a ray (the spheres stand for any objects)

used by the author already for the detection of sound particles in the audience region [6]. Thereby, all the intersection points of the ray with the deviding xy-, yz- and zx-planes of the grid (grid walls) have to be found [1]), the plane found as nearest lets the particle stop. The stop point then serves as a new starting point with a free distance to the next grid wall. If walls are stored to intersect the voxel (of some integer coordinates i,j,k), they are tested for real intersection with the ray (by the point-in polygon-test [6]); if the nearest wall point is within the present voxel, the intersection point is found; else it is stored in a "mailbox" to spare a repeated computation [5]. Due to the fact that most voxels are "empty" or contain maximum 3 walls and due to the regularity of the grid this iteration runs very quickly.

Estimation of the computation time reduction

The smaller the voxels, the smaller the average number K_m of walls contained in a (non empty) voxel, but the higher the average number q_{v0} of empty voxels crossed within a free path length. So, what is now the optimum degree of quantization n_q for a minimum computation time CT_W to find the next wall ? (5)

$$CT_w = CT_0 + q_{v0} \bullet CT_C + K_m \bullet CTU \approx (1 + 0.2 \bullet q_{v0} + K_m) \bullet CTU$$

(*CTU*= computation time unit for the average time to verify a ray-wall intersection needed with classical ray tracing such that the total time to find the next wall is K_0 *CTU*, CT_0=empirical initialization time = ca. 1*CTU*, CT_C = computation time to cross one voxel = ca. 0.2 *CTU*).

q_{v0} is given by the proportion of the mean free path lengths of the room Λ to that of the voxel $\Lambda_R = 2/3 \cdot d$. With $\sigma \approx 0.7$ this is $q_{v0} \approx 0.87 \bullet n_q$ (6).

Now, to estimate K_m , it has to be estimated how many voxels are intersected by at least 1, 2 ,3 or more walls N_1,N_2, N_3 respectively. N_1 is the proportion of the room surface to the voxel´s average cross section $S_V = 2/3 \bullet d^2$ which can be derived to be $N_1 = 10.4 \bullet n_q^2$ (7).

N_2 is the proportion of the total length of all edges of the room $L_{KGes} \approx 2.12 \cdot \sqrt{K_0} \cdot \sqrt{S}$ (8)

(which is valid for the frequent case of side length proportions of roughly 1:2) to Λ_R which is

$$N_2 \approx 8.4 \bullet \sqrt{K_0} \bullet n_q \qquad (9).$$

With an average number of 5 corners per wall (found empirically at typical auditoria) $N_3 \approx 5/3 \cdot K_0$ (10).

Now the average number of walls per non empty voxel is

$$K_m = \frac{n_1 \bullet 1 + n_2 \bullet 2 + n_1 \bullet 3}{n_1 + n_2 + n_3} \qquad (11)$$

where
$$n_3 \approx N_3 \qquad (12)$$
$$n_2 = N_2 - 3 \cdot N_3 \qquad (13)$$
$$n_1 = N_1 - 2 \cdot n_2 - 3 \cdot n_3 = N_1 - 2 \cdot N_2 + 3 \cdot N_3 \qquad (14)$$

are the numbers of voxels containing *exactly* 1,2 or 3 walls. It is assumed that the number of voxels containing more than 3 walls may be neglected.

Insertion of equs. 12-14 with 7,9,10 in equ. 11 yields

$$K_m \approx \left\langle 1 - \frac{0.4\sqrt{K_0}}{n_q} \right\rangle^{-2} \qquad (15)$$

and with equ. 6 inserted into equ. 5 for $\sigma \approx 0.7$: (16)

$$CT_w = (1 + 0.174 \bullet n_q + (\ 1 - \frac{0.4 \bullet \sqrt{K_0}}{n_q}\ )^{-2}) \bullet CTU$$

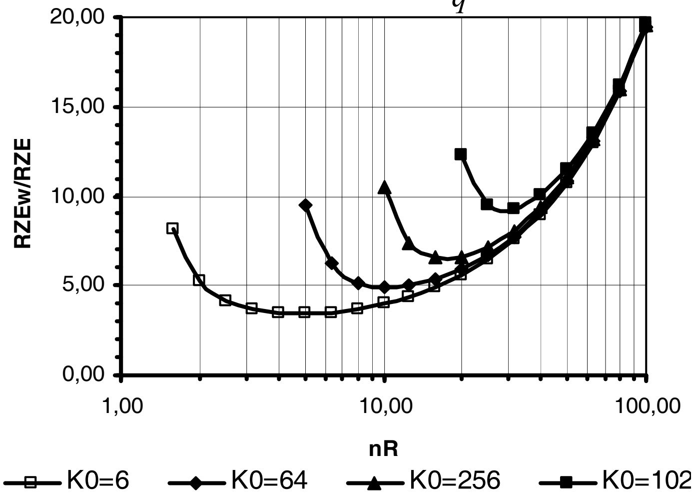

Figure 3: computation time in CTU for finding the next wall as a function of the degree of quantization $n_r = n_q$ with the voxel crossing technique according equ.16 ; parameter K_0= number of walls ; to be compared with K_0 CPU =constant for non-accelerated ray tracing

The minima of these functions would be obtained by derivation with respect to n_q. However, it turns out that near their minima the general assumption of the derivation (no voxels with more than 3 walls) is not fulfilled. In order to choose the voxel size so small (the minima are quite “flat” anyway), that there are only very few with more than 3 walls, obtained if their mean free crossing length Λ_R is much smaller than the mean distances of the room´s corners (edge lengths), an optimum formula for the degree of quantization was found with

$$n_{qO} = 1.5 \bullet \sqrt{\sigma} \bullet \sqrt{K_0} \qquad (17).$$

With this value the average number of walls in a voxel is (by equ. 14) about $K_m \approx 2.2$, typically 88% of the voxels are empty, the numbers n_1, n_2, n_3 become, independently from K_0, 0.5%, 8.8%, 2.6% , less then 0.1% have more than 3 walls, and the formula for the minimum computation time reduces to $CT_w = (3.16 + 0.371 \bullet \sigma^{1.5} \bullet \sqrt{K_0}) \bullet CTU$ (18).

Conclusion

As is can be seen, with the voxel technique, the computation time increases only with the square radix $\sqrt{K_0}$ of the number of walls (instead with K_0). With that, the optimum size of the voxels is (independent from σ)

$$d_O \approx 0.3 \bullet \sqrt{S/K_0} \qquad (19)$$

This is on average about 35% of the mean distance between room corners (edge lengths). It should be pointed out, that this optimum is independent from wavelength.

Theoretically, computation time with rooms of 100, 1000, or 4000 polygons is, with respect to not accelerated ray tracing, reduced by factors of 23,100,240 respectively. (A first implementation of the voxel crossing algorithm in 1990 however showed an effect three times less.) This means empirically, that on modern PCs, computation time for a full room acoustical simulation (impulse responses up to about 10th order of reflection, level accuracy in the order of 0.3dB) even for highly complicated rooms **computation time may be reduced to a few seconds.**

The **storage need** for all the information about the voxel-wall-assignments may be, in the optimum case and with a well adapted storage technique, be estimated by the formula

$$SN_o \approx \left(4 \bullet K_0^{1.5} + 36 \bullet K_0\right) \text{byte} \qquad (20)$$

This effect is achieved to almost the same extent if diffraction is introduced. The performance of the voxel technique, however, is (as any other technique of spatial subdivision) not sufficient to fight against the exponential increase of computation time after introduction of diffraction into ray tracing, for which it originally was investigated.

References

[1] Forsberg, P.-A.: Fully Discrete Ray Tracing. Appl. Acoustics 18 (1985), 393-397

[2] Stephenson,U.; Room Acoustical Design of Concert Halls: Some Recent Improvements on the Sound Particle Simulation Technique; 29th Conference on Acoustics, Czechoslovak Acoustical Society (1990), Strebske Pleso, High Tatra, 259-264

[3] Stephenson,U.M.; Raumakustik-Optimierung: Rechenzeitverkürzung bei der Schallteilchensimulation; In: 17. Tonmeistertagung, Karlsruhe, 1992, Bericht; Saur-Verlag, München 1993

[4] Lehnert,H.; Duerrer,B.; Untersuchungen zur Optimierung des Rechenzeitverhaltens des Strahlverfolgungsverfahrens; In: Fortschritte der Akustik, DAGA 1993, Frankfurt, DPG-GmbH, Bad Honnef, 1993, 227-230

[5] Glassner, A. (ed.) : An Introduction to Ray Tracing; Academic Press Ltd. London, San Diego, 2nd ed. 1990

[6] Stephenson, U.; Eine Schallteilchen-Computer-Simulation zur Berechnung der für die Hörsamkeit in Konzertsälen maßgeben den Parameter. ACUSTICA 59(1985), 1-20.