Acoustics Scattering on Arbitrary Manifold Surfaces

Chandrajit Bajaj *
Dep artment of Computer Scienes,
and Texas Institute of Computational and Applied Mathematics,
University of Texas, Austin, TX 78712

Guoliang Xu †
Institute of Computational Mathematics and Scientific/Engineering Computing,
Chinese Academy of Sciences, Beijing

Joe Warren
Dep artment of Computer Sciene, R ice University, Houston, TX

Abstract

We propose the use of surfac esubdivision as adaptive and higher-order boundary elements for solving a Helmholtz partial differ entialequation to calculate accur ate acoustics scattering on arbitr ary manifolds. Such acoustic transfer functions prove useful for designing and tuning hearing aid devices for he aring impaired individuals. The number of unknowns of the discretize d linear system is the same as that in a line ar element approach. Our results show that the ac cur acy of the subdivision appro ach is much better than that of the line ar element appoach.

Key words: Helmholtz equation, Surface subdivision, Boundary element method.

1 Introduction

We solve a Helmholtz partial differential equation for calculating acoustics scattering on arbitrary manifolds. The acoustics scattering calculation allows the determination of the acoustic pressure on the ear drum, corresponding to different locations of sound and multiple different frequencies. Such acoustic transfer functions pro ve usefulfor designing and tuning hearing aid devices for hearing impaired individuals. The tuning is especially challenging in the case of young children with whom a trial and error approach is not possible.

*Supported in part by NSF grants A CI-9982297, CCR-9988357, a DOE-ASCI grant BD4485-MOID from LLNL/SNL and from grant UCSD 1018140 as part of NSF-NPA CI, In teractive Environments Thrust

†Support in part by NSF of China and National Innovation F und 1770900, Chinese Academy of Sciences

F or the acoustics pressure calculation, the partial differen tial equation (PDE) defined in a 3D domain is reformulated as an in tegral equation over the domain boundary (surface) and then conv erted to a variational form. The problem is finally solved by Galerkin approximations. While a boundary element modeling (BEM) approach is efficient because it converts a 3D problem in an infinite domain into one over a boundary surface in 3D. How ev er,from a n umerical computation point of view, using BEM is rather challenging. The difficulty comes from the evaluation of the singular integration ov er the boundary surface. The singular integration appears when the PDE is con vertedto the in tegral equation and the variational form. The kernel of the singular integration is in the form of $\frac{\partial\Phi(x,y)}{\partial n(y)}$ (see (2.7) and (2.11)), where $\Phi(x,y)=\frac{e^{ik|x-y|}}{4\pi|x-y|}$, $n(y)$ is the surface normal at y. Hence, the kernel is strongly singular as $O(\frac{1}{|x-y|^2})$. This makes the numerical evaluation of the singular in tegration difficult. How ev er, if the domain surface is at least C^1 smooth, the kernel is only weakly singular as $O(\frac{1}{|x-y|})$. Here then, the singular integration is much easier to compute.

We propose using recursiv esubdivision techniques for the solution of boundary element methods. This approach has the following attractive features:

1. Both the domain surface and the pressure function on the surface are defined using the same recursive subdivision technique, the function and surface are C^2 except for some finite set of points (extraordinary points) where it is only C^1 .
2. Both the domain surface and the pressure function are defined in a uniform fashion. This not only yields ease of implementation, but also make

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 **IEEE**

the approximations of the domain surface and the pressure function consistent.

3. Though the number of unknowns of the discretized system is the same as that of linear element method, all our experiments show that the accuracy of the subdivision approach is much better than that of the linear element approach.

We should point out that though we focus our attention specifically on the Helmholtz equation, the approach could be applied to other types of PDEs as well, especially, the problems that require smooth domains.

The rest of the paper is organized as follows. Section 2 reviews the mathematical formulation of the acoustics scattering problem, both in the PDE form and its variational counterpart. Section 3 discretizes the variational problem by a Galerkin approximation in a general functional space. Then in section 4, we describe a known recursive subdivision scheme over triangular meshes for modeling the domain boundary surface and any function on the surface. In section 5, we compute the stiffness matrix including the singular integration evaluation in the space defined by the limit solutions of the recursive subdivision. The final section concludes the paper with examples and comparison with the linear element solution.

2 Mathematical Formulation

In this section, we review the mathematical formulation of the acoustics scattering problem. The problem is defined initially by a partial differential equation (PDE), and then it is reformulated as an integral equation and finally it is converted to its variational counterpart. The interested reader is referred to [3, 14] for more details.

2.1 Partial Differential Equation

Let $\Omega \subset \mathbb{R}^3$ denote a bounded domain with boundary surface Γ (see Fig 2.1). We require Γ be a smooth closed surface. Given an incident pressure field p^{inc}

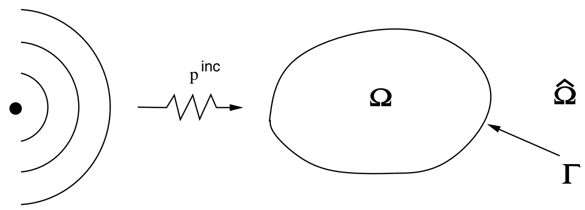


Fig 2.1: The model of acoustics scattering.

in $\mathbb{R}^3$, we wish to determine a (complex-valued) total pressure function

$$p = p^{inc} + p^s \quad \text{in} \quad \hat{\Omega} = \mathbb{R}^3 - \Omega \tag{2.1}$$

satisfying the following Helmholtz equation

$$\Delta p + k^2 p = 0 \quad \text{in} \quad \hat{\Omega}, \tag{2.2}$$

with a rigid boundary condition on Γ

$$\frac{\partial p}{\partial n} = 0, \tag{2.3}$$

and the scattered pressure p^s function satisfying the Sommerfeld radiation condition

$$\left| \frac{\partial p^s}{\partial R} - ikp^s \right| = O\left(\frac{1}{R^2}\right) \quad \text{for} \quad R \to \infty, \tag{2.4}$$

where i is the imaginary unit, $k = 2\pi f/c$ is the wave number (f is frequency and c is sound velocity), R is the distance from the origin and n is the outward unit normal for the domain boundary Γ.

2.2 Integral Equation

Since the PDE (2.2) is defined in the infinite domain $\hat{\Omega}$ and the goal is to find the solution on the boundary Γ, converting the PDE to an integral equation that is valid on Γ, should be naturally more efficient. Following [3], we replace Helmholtz equation (2.2) and the Sommerfeld radiation condition (2.4) with the equivalent Burton-Miller boundary integral equation

$$\begin{aligned} \frac{1}{2}p - Cp + A\frac{\partial p}{\partial n} + \frac{i}{k}\left(\frac{1}{2}\frac{\partial p}{\partial n} + B\frac{\partial p}{\partial n} + Dp\right) \\ = p^{inc} + \frac{i}{k}\frac{\partial p^{inc}}{\partial n}, \end{aligned} \tag{2.5}$$

where the boundary integral operators are defined as follows:
The single layer potential

$$Ap(x) = \int_\Gamma \Phi(x,y)p(y)dS(y). \tag{2.6}$$

The double layer potential

$$Cp(x) = \int_\Gamma \frac{\partial \Phi(x,y)}{\partial n(y)} p(y)dS(y). \tag{2.7}$$

The adjoint double layer potential

$$Bp(x) = \int_\Gamma \frac{\partial \Phi(x,y)}{\partial n(x)} p(y)dS(y). \tag{2.8}$$

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 IEEE

The hypersingular operator

$$Dp(x) = \int_\Gamma \frac{\partial^2 \Phi(x,y)}{\partial n(x) \partial n(y)} p(y) dS(y). \qquad (2.9)$$

Where $\Phi(x,y) = \Phi(r) = \frac{1}{4\pi}\frac{e^{ikr}}{r}$, with $r = |x-y|$ is the fundamental solution to the Helmholtz equation and additionally with derivatives:

$$\frac{\partial \Phi(x,y)}{\partial n(x)} = \Phi'(r)\frac{\partial r}{\partial n(x)}, \qquad \frac{\partial \Phi(x,y)}{\partial n(y)} = \Phi'(r)\frac{\partial r}{\partial n(y)},$$
$$\frac{\partial^2 \Phi(x,y)}{\partial n(x)\partial n(y)} = \Phi''(r)\frac{\partial r}{\partial n(x)}\frac{\partial r}{\partial n(y)} + \Phi'(r)\frac{\partial^2 r}{\partial n(x)\partial n(y)}.$$

The integration (2.6) exists in the usual Lebesgue integral sense. The integrals of (2.7) and (2.8) are defined in the Cauchy Principle Value (CPV) sense, and the integral in (2.9) is defined using the notion of the Hadamard finite part integral. For a smooth domain Γ, the CPV integrals reduce to the Lebesgue integral as well.

2.3 Variational Formulation

The integral equation can next be converted to its variational form and from there a linear problem is obtained via a Galerkin approximation. To construct the variational form, we multiply (2.5) by a test function $\bar{q}$, integrate once more over the boundary Γ with respect to the variable x, and integrate the hypersingular term by parts moving one derivative to the test function, to arrive at the identity

$$d(p,q) = l(q) \qquad (2.10)$$

for any admissible q. Here the sesquilinear and antilinear forms are defined as follows:

$$\begin{aligned} d(p,q) = &\frac{1}{2}\int_\Gamma p(x)\bar{q}(x)dS(x) \\ &- \int_\Gamma\int_\Gamma \frac{\partial\Phi(x,y)}{\partial n(y)} p(y)\bar{q}(x)dS(y)dS(x) \\ &+ \frac{i}{k}\left[\int_\Gamma\int_\Gamma \Phi(x,y)\left(rot_y p(y)\right)^T rot_x \bar{q}(x)dS(y)dS(x)\right. \\ &\left.- k^2\int_\Gamma\int_\Gamma \Phi(x,y)n(x)^T n(y)p(y)\bar{q}(x)dS(y)dS(x)\right] \quad (2.11) \end{aligned}$$

$$\begin{aligned} l(q) = &\int_\Gamma p^{inc}(x)\bar{q}(x)dS(x) \\ &+ \frac{i}{k}\int_\Gamma \frac{\partial p^{inc}(x)}{\partial n(x)}\bar{q}(x)dS(x) \quad (2.12) \end{aligned}$$

with $rot_x p = \nabla p \times n(x)$. We may reduce (2.10) to the standard variational formulation

$$\begin{cases} \text{Find } \ p \in H^{\frac{1}{2}}(\Gamma) \text{ such that} \\ d(p,q) = l(q), \quad \forall q \in H^{\frac{1}{2}}(\Gamma), \end{cases} \qquad (2.13)$$

where $H^{\frac{1}{2}}(\Gamma)$ is the Sobolev space of order $1/2$ for functions defined on the boundary Γ.

3 Galerkin Approximation

Based on the variational formulation, the usual Galerkin approximation can be applied. Given a set of basis functions of a finite dimensional sub-space $V^h \subset H^{\frac{1}{2}}(\Gamma)$: $\phi_i(x)$, $i = 1,\cdots,N$, we introduce the following form approximations of the total pressure $p(x)$

$$p^h(x) = \sum_{i=1}^N p_i\phi_i(x), \qquad (3.1)$$

where $p_i \in \mathbb{C}$ are complex unknowns to be determined. In the variational problem (2.13), taking $p = p^h$, $q = \phi_i$ for $i = 1,\cdots,N$, we obtain the linear system

$$\sum_{i=1}^N d_{ik}p_i = l_k, \quad k = 1,\cdots,N, \qquad (3.2)$$

where

$$\begin{aligned} d_{ik} &= d(\phi_i,\phi_k), \quad i,k = 1,\cdots,N, \\ l_k &= l(\phi_k), \qquad\quad k = 1,\cdots,N \end{aligned} \qquad (3.3)$$

are constants. Note that, in contrast to the finite element case, the coefficient matrix of (3.2) is dense even though ϕ_i is locally supported. Solving system (3.2), we are able to compute the required approximate solution. Hence the solution to the Helmholtz equation under the given boundary and radiation conditions reduces to the computation of the coefficient matrix and the right-handed side. Let surface Γ be expressed as the union of triangular patches (how to define these patches is the topic of the next section) $e_i, i = 1,\cdots,M$. The single and double integrations $\int_\Gamma \cdot\, dS(x)$ and $\int_\Gamma\int_\Gamma \cdot\, dS(y)dS(x)$ in (2.11)-(2.12) could be expressed as

$$\sum_{s=1}^M \int_{e_s} \cdot\, dS(y)dS(x), \ \sum_{s=1}^M\sum_{t=1}^M \int_{e_s}\int_{e_t} \cdot\, dS(y)dS(x) \qquad (3.4)$$

respectively. The integrals in (3.4) are computed by a certain numerical quadrature rule. Hence several surface points and normals in each patch e_i need to be evaluated. Therefore, our remaining questions are: (a) What patches should we use to represent the boundary surface Γ? (b) What is a suitable finite-dimensional pressure function space V^h and its compactly supported basis functions ϕ_i?

The surface representation should facilitate the evaluation of the surface points and surface normals. The

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 IEEE

basis functions are required to be local support to simplify the computation. In [14] triangular C^1 A-patches [1], were used to smoothly model the domain boundary. In this paper, Γ shall be defined by the limit surface of a known recursive subdivision scheme. The basis functions ϕ_i for the pressure function are defined by the same recursive subdivision rule. Discussions on the selected subdivision scheme will be the topic of the next section.

In the following, we shall assume the input for Γ is a triangular mesh consisting of M triangles $\{T_i\}_{i=1}^{M}$ and N vertices $\{v_i\}_{i=1}^{N}$. For each T_i, we shall construct a curved smooth triangular surface patch e_i, which interpolates the three vertices of T_i and the union of all e_i is a smooth representation of Γ.

4 Recursive Subdivision of Triangular Meshes

We shall discretize the variational problem (2.13) in a function space which is defined by the limit of Loop's recursive subdivision. This section describes only the relevant results on surface subdivision. It will be clear soon that these results are valid on the subdivision of functions defined on surfaces.

Subdivision schemes generate smooth surfaces via a limit procedure of an iterative refinement starting from an initial mesh which serves as the control mesh of the limit surface. Several subdivision schemes for generating smooth surfaces have been proposed. Some of them are interpolatory, i.e., the vertex positions of the coarse mesh are fixed, and only the newly added vertex positions need to be computed (see e.g., [7] for quadrilateral meshes, [5, 16] for triangular meshes), while others are approximatory (see e.g., [2, 4] for quadrilateral meshes, [8] for triangular meshes, [9] for general polyhedra). These approximatory subdivision schemes compute both the old and new vertex positions at each refinement step. Generally speaking, approximatory schemes produce better quality surfaces than those produced by interpolatory schemes. Hence, in this work, we shall use an approximating scheme for triangular meshes proposed by Loop [8]. This scheme produces C^2 limit surfaces except at a finite number of isolated (extraordinary) points where the surface is C^1 ([10]).

For Loop's scheme, a closed form and fast method exists for evaluating the limit surfaces and its normals at any parameter value (see [13]), especially needed for the numerical computation of the area-integrals. Of course, any other scheme that supports fast exact evaluation can be used here. For instance, Catmull-Clark's scheme (see [12]) for quadrilateral meshes can serve us equally well. The choice of triangular or quadrilateral subdivision scheme can be left to favour the type of the input domain representation.

4.1 Loop's Subdivision Scheme

In Loop's subdivision scheme, the initial control mesh and the subsequent refined meshes consist of only triangles. In a refinement step, each triangle is subdivided linearly into 4 sub-triangles. Then all the vertex positions of the refined mesh is computed as the weighted average of the vertex positions of the unrefined mesh. Consider a vertex x_0^k at level k with neighbor vertices x_i^k for $i = 1, \cdots, n$ (see Fig 4.1), where n is the valence of vertex x_0^k. The coordinates of the newly generated vertices x_i^{k+1} on the edges of the previous mesh are computed as

$$x_i^{k+1} = \frac{3x_0^k + 3x_i^k + x_{i-1}^k + x_{i+1}^k}{8}, \quad i = 1, \cdots, n, \quad (4.1)$$

where index i is to be understood modulo n. The old vertices get new positions according to

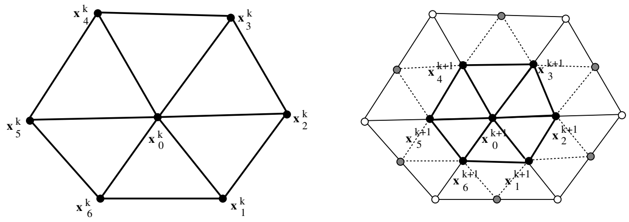


Fig 4.1: Refinement of triangular mesh around a vertex.

$$x_0^{k+1} = (1 - na)x_0^k + a\left(x_1^k + x_2^k + \cdots + x_n^k\right), \quad (4.2)$$

where $a = \frac{1}{n}\left[\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4}cos\frac{2\pi}{n}\right)^2\right]$. Note that all newly generated vertices have a valence of 6, while the vertices inherited from the original mesh at level zero may have a valence other than 6. The former case is refereed to as *ordinary* and the latter case is referred as *extraordinary*. The limit surface of Loop's subdivision is C^2 everywhere except at the extraordinary points where it is C^1 [8].

4.2 The Limit Surface Corresponding to Vertices

Let x_0^0 be a vertex with x_i^0, $i = 1, \cdots, n$, being the 1-ring neighbor vertices of the initial mesh. Then all these vertices converge to a single position

$$(a_0^0)^T = (1 - nl)x_0^0 + l\sum_{i=1}^{n} x_i^0, \quad l = 1/(n + 3/8a) \quad (4.3)$$

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 IEEE

as the subdivision step goes to infinity. This means that we can evaluate the limit position of the surface at any finite subdivision level and at any vertex by simply averaging the vertex and its neighbors. The surface tangents corresponding to the edges $[x_0^0 x_j^0]$ around x_0^0 are given by the following formula

$$t_{j+1} = cos\left(\frac{2\pi j}{n}\right) a_1^0 + sin\left(\frac{2\pi j}{n}\right) a_{n-1}^0$$

for $j = 0, \cdots, n-1$, where

$$\begin{aligned} a_1^0 &= \frac{2}{n}\sum_{i=0}^{n-1} cos\left(\frac{2\pi i}{n}\right) x_{i+1}^0, \\ a_{n-1}^0 &= \frac{2}{n}\sum_{i=0}^{n-1} sin\left(\frac{2\pi i}{n}\right) x_{i+1}^0. \end{aligned}$$

4.3 Evaluation of Regular Surface Patches

To obtain a local parameterization of the limit surface for each of the triangles in the initial control mesh, we choose (ξ_1, ξ_2) as two of the barycentric coordinates (ξ_0, ξ_1, ξ_2) and define T as

$$T = \{(\xi_1, \xi_2) \in \mathbb{R}^2 : \xi_1 \geq 0, \xi_2 \geq 0, \xi_1 + \xi_2 \leq 1\}.$$

The triangle T in the (ξ_1, ξ_2)-plane may be used as a master element domain. Consider a generic triangle in the mesh and introduce a local numbering of vertices lying in its immediate 1-ring neighborhood (see Fig 4.2). If all its vertices have a valence of 6, the resulting patch of the limit surface is exactly described by a single quartic box-spline patch, for which an explicit closed form exists [13]. We refer to such a patch as *regular*. A regular patch is controlled by 12 basis functions:

$$x(\xi_1, \xi_2) = \sum_{i=1}^{12} N_i(\xi_1, \xi_2) x_i, \tag{4.4}$$

where the label i refers to the local numbering of the vertices that is shown in Fig 4.2. The surface within the shaded triangle in this figure is defined by the 12 local control vertices. The basis N_i are given as follows (see [13]):

$$\begin{aligned} N_1 &= \tfrac{1}{12}(\xi_0^4 + 2\xi_0^3\xi_1), \\ N_2 &= \tfrac{1}{12}(\xi_0^4 + 2\xi_0^3\xi_2), \\ N_3 &= \tfrac{1}{12}\left[\xi_0^4 + \xi_1^4 + 6\xi_0^3\xi_1 + 6\xi_0\xi_1^3 + 12\xi_0^2\xi_1^2 \right. \\ &\quad \left. + (2\xi_0^3 + 2\xi_1^3 + 6\xi_0^2\xi_1 + 6\xi_0\xi_1^2)\xi_2\right], \\ N_4 &= \tfrac{1}{12}[6\xi_0^4 + 24\xi_0^3(\xi_1+\xi_2) \\ &\quad + \xi_0^2(24\xi_1^2 + 60\xi_1\xi_2 + 24\xi_2^2) \\ &\quad + \xi_0(8\xi_1^3 + 36\xi_1^2\xi_2 + 36\xi_1\xi_2^2 + 8\xi_2^3) \\ &\quad + (\xi_1^4 + 6\xi_1^3\xi_2 + 12\xi_1^2\xi_2^2 + 6\xi_1\xi_2^3 + \xi_2^4)], \end{aligned} \tag{4.5}$$

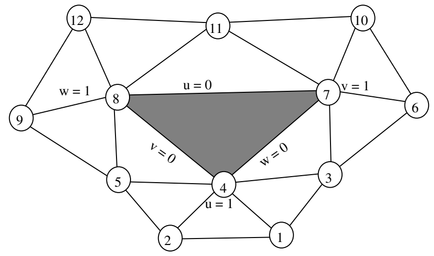


Fig 4.2: The vertex numbering of a regular patch with 12 control points. A regular patch is defined over the shaded triangle.

where (ξ_0, ξ_1, ξ_2) are barycentric coordinates of the triangle with vertices numbered as $4, 7, 8$, and $\xi_0 = 1 - \xi_1 - \xi_2$. Other basis functions are similarly defined. For example, replacing (ξ_0, ξ_1, ξ_2) by (ξ_1, ξ_2, ξ_0) in N_1, N_2, N_3, N_4, we get N_{10}, N_6, N_{11}, N_7. Replacing (ξ_0, ξ_1, ξ_2) by (ξ_2, ξ_0, ξ_1) we get N_9, N_{12}, N_5, N_8.

4.4 Evaluation of Irregular Surface Patches

If a triangle is irregular, i.e., at least one of its vertices has a valence other than 6, the resulting patch is not a quartic box spline. We assume extraordinary vertices are isolated, i.e., there is no edge in the control mesh such that both its vertices are extraordinary. This assumption can be met by subdividing the mesh once. Under this assumption, any irregular patch has only one extraordinary vertex. For the evaluation of irregular patches, we use the scheme proposed by Stam [13]. In this scheme the mesh needs to be subdivided repeatedly until the parameter values of interest are interior to a regular patch. We now summarize the central idea of Stam's scheme. First, it is easy to see that each subdivision of an irregular patch produces three regular patches and one irregular patch (see Fig 4.3).

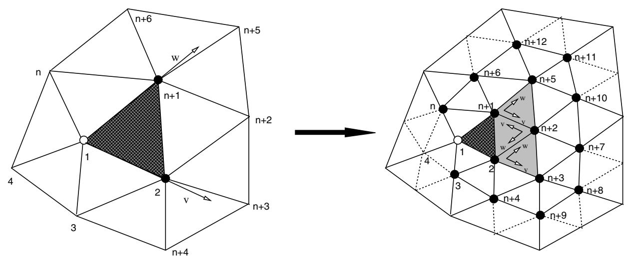


Fig 4.3: The vertex with empty circle is extraordinary. After one subdivision step, the irregular patch (dark shaded triangle) is split into one irregular patch (smaller dark shaded triangle) and three regular patches (unshaded parts).

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 IEEE

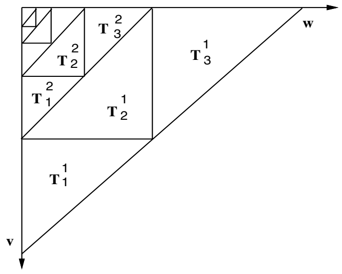


Fig 4.4: Refinement in the parametric space, where $(u,v,w) = (\xi_0, \xi_1, \xi_2)$ is the barycentric coordinates of the triangle.

Repeated subdivision of the irregular patch produce a sequence of regular patches. The surface patch is piecewise parameterized as shown in Fig 4.4. The subdomains T_j^k are given as follows:

$$\begin{aligned}
T_1^k &= \{(\xi_1,\xi_2) : \xi_1 \in [\tfrac{1}{2^k}, \tfrac{1}{2^{k-1}}], & \xi_2 \in [0, \tfrac{1}{2^{k-1}} - \xi_1]\}, \\
T_2^k &= \{(\xi_1,\xi_2) : \xi_1 \in [0, \tfrac{1}{2^k}], & \xi_2 \in [\tfrac{1}{2^k} - \xi_1, \tfrac{1}{2^k}]\}, \\
T_3^k &= \{(\xi_1,\xi_2) : \xi_1 \in [0, \tfrac{1}{2^k}], & \xi_2 \in [\tfrac{1}{2^k}, \tfrac{1}{2^{k-1}} - \xi_1]\}.
\end{aligned}$$

These subdomains are mapped onto T by the transform

$$\begin{aligned}
t_{k,1}(\xi_1,\xi_2) &= (2^k\xi_1 - 1, 2^n\xi_2), & (\xi_1,\xi_2) \in T_1^k, \\
t_{k,2}(\xi_1,\xi_2) &= (1 - 2^k\xi_1, 1 - 2^k\xi_2), & (\xi_1,\xi_2) \in T_2^k, \\
t_{k,3}(\xi_1,\xi_2) &= (2^k\xi_1, 2^k\xi_2 - 1), & (\xi_1,\xi_2) \in T_3^k.
\end{aligned}$$

Hence T_j^k form a tiling of T except for the point $(\xi_1,\xi_2) = (0,0)$. The surface patch is then defined by its restriction to each triangle

$$x(\xi_1,\xi_2)|_{T_j^k} = \sum_{i=1}^{12} x_i^{k,j} N_i(t_{k,j}(\xi_1,\xi_2)) \tag{4.6}$$

for $j = 1,2,3$; $k = 1,2,\cdots$, where $x_i^{k,j}$ are the properly chosen 12 control vertices around the irregular patch at the level k that define a regular surface patch. Using the vertex numbering and local coordinate system shown in Fig 4.3, it is easy to see that the three sets of control vertices are

$$\begin{aligned}
\{x_i^{k,1}\}_{i=1}^{12} &= [x_3^k, x_1^k, x_{n+4}^k, x_2^k, x_{n+1}^k, x_{n+9}^k, x_{n+3}^k, \\
&\quad x_{n+2}^k, x_{n+5}^k, x_{n+8}^k, x_{n+7}^k, x_{n+10}^k], \\
\{x_i^{k,2}\}_{i=1}^{12} &= [x_{n+7}^k, x_{n+10}^k, x_{n+3}^k, x_{n+2}^k, x_{n+5}^k, x_{n+4}^k, \\
&\quad x_2^k, x_{n+1}^k, x_{n+6}^k, x_3^k, x_1^k, x_n^k], \\
\{x_i^{k,3}\}_{i=1}^{12} &= [x_1^k, x_n^k, x_2^k, x_{n+1}^k, x_{n+6}^k, x_{n+3}^k, x_{n+2}^k, \\
&\quad x_{n+5}^k, x_{n+12}^k, x_{n+7}^k, x_{n+10}^k, x_{n+11}^k].
\end{aligned}$$

Hence, the main task is to compute these control vertices. As usual, the subdivision around an irregular patch is formulated as a linear transform from the level $(k-1)$, 1-ring vertices of the irregular patch to the related level k vertices, i.e.,

$$\begin{aligned}
X^k &= AX^{k-1} = \cdots = A^k X^0, \\
\tilde{X}^{k+1} &= \tilde{A}X^k = \tilde{A}A^k X^0,
\end{aligned}$$

where

$$\begin{aligned}
X^k &= [x_1^k, \cdots, x_{n+6}^k]^T, \\
\tilde{X}^k &= [x_1^k, \cdots, x_{n+6}^k, x_{n+7}^k, \cdots, x_{n+12}^k]^T,
\end{aligned}$$

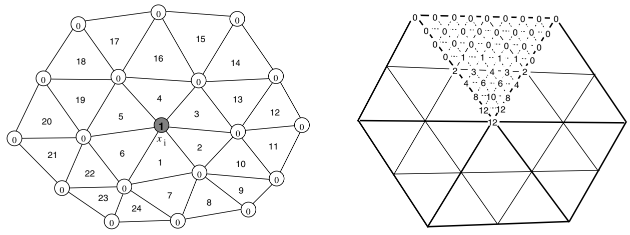
and A and $\tilde{A}$ are defined by the subdivision rules. Hence, $k+1$ subdivisions lead to the computation of A^k. When k is large, the computation can be very time consuming. A novel ideaproposed by Jos Stam is to use the Jordan canonical form $A = SJS^{-1}$. The computation of A^k reduces to the computation of J^k, which makes the cost of the irregular patch computation nearly independent of k and hence very efficient. The beauty of the scheme is that explicit forms of both S and J exist. We refer to [13] for details.

4.5 Basis Functions and Classification of Patches

For each vertex x_i of a control mesh Γ_d, we shall associate with a hat basis function ϕ_i, where ϕ_i is defined by the limit of the Loop's subdivision, with zero control values everywhere except at x_i where it is one (see Fig. 4.5.a). Hence the support of ϕ_i is local and it covers the 2-ring neighborhood of vertex x_i. Let e_j, $j = 1,\cdots,m_i$ be the 2-ring neighborhood elements. Then if e_j is regular, the explicit box-spline expression as in (4.4) exists for ϕ_i on e_j. Using (4.5), we could derive the BB-form coefficients for basis ϕ_i (see Fig. 4.5.b). All these coefficients have a factor $\frac{1}{24}$. Hence, the function value at x_i is $\frac{1}{2}$. These expressions could be used to evaluate ϕ_i in forming the linear system (3.2). If e_i is irregular, local subdivision, as described in §4.4, is needed around e_i until the parameter values of interest are interior to a regular patch.

Using the basis $\{\phi_i\}$, the limit surface of Loop's subdivision is expressed as $\Gamma = \sum x_i\phi_i(x)$. However, each triangular surface patch of Γ is defined locally by only a few related basis functions, since the support of the basis functions is compact. For a triangle $[x_i x_i x_k]$, the related basis functions that defines the surface patch over the triangle are uniquely determined by the valences n_i, n_j and n_k, here n_i, n_j and n_k are the valences of vertex x_i, x_j and x_k, respectively. Hence, two triangles that have the same valence for each of the three vertices, will have the same set of related basis functions. To reduce the computation costs of evaluating these functions in the numerical integration, triangles are classified into categories according their vertex

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 IEEE



(a) (b)

Fig 4.5: (a). Numbered 2-ring neighborhood elements of vertex x_i. The vertex numbers in circles are the control coefficients which also define the basis ϕ_i. (b). The quartic Bézier coefficients (each has a factor 1/24) of the basis function. The coefficients on the other five macro-triangles are obtained by rotating the top macro-triangle around the center, to the other five positions.

valences. All members in one category will have the same vertex valences, hence the same set of related basis functions. For one category of patches, we only need to evaluate the basis functions once. Using depth first search, the classification can be computed within linear time.

4.6 The Initial Control Mesh

Suppose we are given a surface triangulation for Γ. Since Loop's subdivision scheme is not interpolatory, the limit surface of the subdivision starting from any given triangulation will not interpolate the original vertices. Therefore, we need to define an initial control mesh so that the limit surface of the subdivision, starting from this control mesh, interpolates the vertices of the input triangulation. Using formula (4.3), we have

$$(1 - n_i l_i)x_i + l_i \sum_{j=1}^{n_i} x_{k_j} = v_i, \quad i = 1, \cdots, N, \qquad (4.7)$$

where x_{k_j} is the 1-ring neighborhood of x_i, v_i is the input vertex that is on the boundary surface Γ, x_i are the unknown positions to be determined. n_i is the valance of vertex x_i, and $l_i = 1/(n_i + 3/8a)$. Solving the system (4.7), we get x_i's. Equation (4.7) is $N \times N$ system, where N may be large. The linear system is sparse and one may solve the system by an iterative method, e.g., Jacobi iterative method. We even do not need to store the matrix since its elements could be easily computed during the iteration. A good initial value of x_i^0 for the iteration could be v_i.

5 The Linear System

The pressure function to be determined on Γ is defined by the limit of Loop's subdivision scheme, that is the same recursive scheme for constructing the boundary surface Γ. Letting the basis of the limit function at vertex x_i be $\phi_i(x)$, we have the pressure function

$$p^h(x) = \sum_{i=1}^{N} p_i \phi_i(x) \qquad (5.1)$$

where p_i are the unknowns to be determined. Then the linear system (3.2) is generated using the following C style pseudo code:

```
for (k = 1, k <= N, k++) {
    for (i = 1, i <= N, i++) {
        d_ik = 0;
        for (s = 1, s <= m_k, s++) {
            for (t = 1, t <= m_i, t++) {
                d_ik = d_ik + D_{e_s e_t}(phi_i, phi_k);
            }
        }
    }
    l_k = 0;
    for (s = 1, s <= m_k, s++) {
        l_k = l_k + l_{e_s}(phi_k);
    }
}
```

where $D_{e_s e_t}(p,q)$ is defined as $d(p,q)$, but the double integration domain is replaced by (e_s, e_t) for (x,y) variables, respectively. In a similar fashion, $l_{e_s}(p)$ can be determined.

5.1 Numerical Integration

It follows from the sesquilinear and antilinear forms (2.11)–(2.12) that we need to handle two types of integrals. The first type is where the integrand is a bounded function (non-singularity). These integrals appear in (2.11)–(2.12) as single integrals or outer integrals of the double integrals. The second type is where the integrand has a singularity due to the function Φ, which includes the factor $\frac{1}{\|x-y\|}$. These integrals appear in (2.11) as inner integrations.

Nonsingular integration We have mentioned that the number of unknowns of our BEM is the same as in the linear element approach. It has been shown that the space spanned by Loop's basis functions has linear accuracy [15]. Hence the full order of the approximation error of the BEM solution is $O(h^2)$, where h is the mesh size which could be defined as the maximal length

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 IEEE

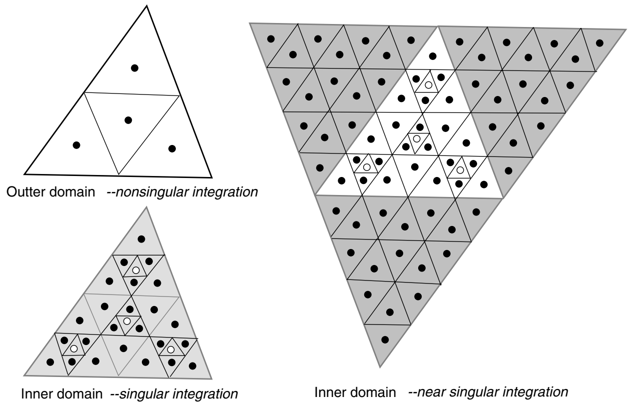


Fig 5.1: An adaptive mesh around singular and nearly singular points and linear numerical integration for $k = 1$, $p = 1$, $s = 2$. The dots are the integration points.

of the edges. For the nonsingular integration, the integration over a triangular surface patch is computed by subdividing the patch uniformly into 4^k sub-patches. Over each sub-patch, a one-point Gauss quadrature rule is used. It is well known that the one-point Gauss quadrature rule has error of the order $O(4^{-k}h^2)$ over a triangle that has size $2^{-k}h$. We subdivide the patch k times to make the integration have accuracy better than that could be achieved by the BEM, so that the error of the BEM solution is not controlled by the error of numerical integration. Another reason we use a one-point Gauss quadrature rule based on the uniform partition of the domain triangle is that the uniform node distribution of the quadrature rule makes the integrand of the near-singular integration behavior better (see next paragraph).

Singular Integration. Note that a surface point x on e_i for an outer integration becomes a singular point for an inner integration, if the point x is located on the domain e_j at the inner integration. Otherwise, the inner integration is not singular. However, if the point x for an outer integration is near the domain e_j, the integrand of the inner integration is nearly singular since $\|x - y\|$ could be small for $y \in e_j$.

For the singular and nearly singular integration, we adopt an adaptive integration strategy (see Fig 5.1). That is a fine adaptive mesh around the singular point is created by repeated subdivision. The density of the mesh increases linearly towards the singular points. Over each sub-element of the adaptive mesh, linear integration or q-version[1] adaptive Gauss integration could be used (see [11]). In the q-version adaptive integration, the algebraic precision is linearly decreasing towards the singular point with slope $\mu \approx 1$. Schwab [11] has proved that such a technique has exponential convergence rate $O(exp(-bN^{1/3}))$, where N is the number of integrand evaluations and $b > 0$ is a constant.

We have experimented with a set of numerical integration schemes in the q-version over triangles. These include one point, three points, four points, six points and seven points rules. Table 5.1 summarizes these rules with coordinates, weights and algebraic precision. Table 5.2 lists the errors of the solution of the BEM for $k = 1, 2, 3$, $q = 1, \cdots, 5$ and $s = 1, \cdots, 7$. The domain surface is the unit sphere discretized as 1280 triangular patches. $p^{inc} = e^{2xi}$. For such a domain and the given p^{inc}, analytic solution for the total pressure distribution is available (see [6]). The errors in the table are defined by $\sqrt{\sum |\tilde{p}_i - p_i|^2/n}$, where $\tilde{p}_i$ and p_i are the computed and the exact solutions, respectively, at the i-th vertex, and n is the number of vertices. It can be seen from the tables that the errors become stable when $s \geq 6$ and $q \geq 4$ for each k. For $k \geq 3$, the computation is very intensive. Hence, in general, we choose $(k, q, s) = (2, 4, 6)$.

6 Comparison and Conclusions

To show the proposed method is correct and efficient, we compute the pressure function p for a sphere domain and a planar wave $p^{inc} = e^{ikx}$. For a sphere domain and this p^{inc}, an analytic solution is available (see [6]). Table 6.1 gives the errors of the computed solutions to the exact solution, for different surface resolutions (32, 128, 512, 2048 triangles) of the unit sphere. In this table, errors are computed by using both Loop basis functions and linear basis functions in our Galerkin approximation. The errors show that the approach based on Loop basis functions is much better than that of the linear basis functions.

We also compared the computation times of our method with that of linear elements for the same number of unknowns. However, for each surface patch, the linear element approach has fewer related non-zero basis functions than Loop basis functions. Hence, computing the stiffness matrix for the linear element approach uses somewhat less computation time. The last column of Table 6.1 shows the computation times for computing the stiffness matrix and for solving the linear system for both linear and Loop basis function approaches. These computations were conducted on a SGI Onyx2, using a single R12k processor.

Finally, we present in Fig 6.1 the computed acoustics pressure distribution over a human head. The left figure is the geometric model of the head. The middle and the right figures are the iso-contour plot of the

[1] Since p = pressure, we substitute the popular terminology for higher order methods from p-version to q-version

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 **IEEE**

Table 5.1: Numerical Integration over triangles. $(1 - v_i - w_i, v_i, w_i)$ are the barycentric coordinates of the nodes. W_i are the summation weight factors. The second row q represents the algebraic precision.

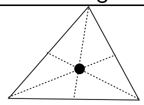
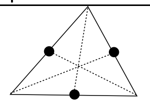
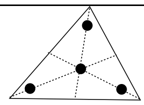
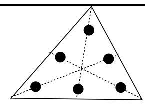
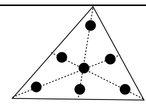
					
q	1	2	3	4	5
v_1	0.3333333333	0.0	0.1333333333	0.8168475729	0.05961587
v_2		0.5	0.1333333333	0.0915762135	0.47014206
v_3		0.5	0.7333333333	0.0915762135	0.47014206
v_4			0.3333333333	0.1081030181	0.79742699
v_5				0.4459484909	0.10128651
v_6				0.4459484909	0.10128651
v_7					0.33333333
w_1	0.3333333333	0.5	0.7333333333	0.0915762135	0.47014206
w_2		0.0	0.1333333333	0.8168475729	0.05961587
w_3		0.5	0.1333333333	0.0915762135	0.47014206
w_4			0.3333333333	0.4459484909	0.10128651
w_5				0.1081030181	0.79742699
w_6				0.4459484909	0.10128651
w_7					0.33333333
W_1	1.0	0.3333333333	0.5208333333	0.1099517436	0.13239415
W_2		0.3333333333	0.5208333333	0.1099517436	0.13239415
W_3		0.3333333333	0.5208333333	0.1099517436	0.13239415
W_4			-0.5625	0.2233815896	0.12593918
W_5				0.2233815896	0.12593918
W_6				0.2233815896	0.12593918
W_7					0.225

Table 5.2: Errors of BEM solutions for $k = 1, 2, 3$, $q = 1, \cdots 5$ and $s = 1, \cdots, 7$.

k = 1	s=1	s =2	s =3	s = 4	s = 5	s = 6	s = 7
q=1	0.0236	0.0116	0.00692	0.00492	0.00405	0.00401	0.00369
q=2	0.0235	0.0115	0.00675	0.00476	0.00391	0.00372	0.00343
q=3	0.0232	0.0112	0.00655	0.00460	0.00376	0.00338	0.00352
q=4	0.0226	0.0105	0.00580	0.00380	0.00298	0.00267	0.00260
q=5	0.0226	0.0105	0.00578	0.00386	0.00308	0.00279	0.00254
k = 2	**s=1**	**s =2**	**s =3**	**s = 4**	**s = 5**	**s = 6**	**s = 7**
q=1	0.0175	0.00694	0.00350	0.00226	0.00190	0.00202	0.00216
q=2	0.0174	0.00684	0.00339	0.00211	0.00163	0.00156	0.00128
q=3	0.0172	0.00669	0.00325	0.00203	0.00154	0.00143	0.00180
q=4	0.0169	0.00633	0.00285	0.00157	0.00113	0.00101	0.00159
q=5	0.0169	0.00632	0.00288	0.00167	0.00125	0.00106	0.00100
k = 3	**s=1**	**s =2**	**s =3**	**s = 4**	**s = 5**	**s = 6**	**s = 7**
q=1	0.0149	0.00491	0.00216	0.00137	0.00127	0.00103	0.00144
q=2	0.0147	0.00487	0.00209	0.00121	0.00109	0.00082	0.00150
q=3	0.0146	0.00475	0.00200	0.00107	0.00091	0.00074	0.00156
q=4	0.0143	0.00460	0.00174	0.00094	0.00083	0.00069	0.00063
q=5	0.0145	0.00460	0.00178	0.00097	0.00077	0.00076	0.00068

real and imaginary parts of the pressure function. The smooth iso-contours exhibit the pressure functions over the head are smooth.

Acknowledgement: We wish to thank Dr. Tim Walsh for several discussions.

References

[1] C. Bajaj, J. Chen, and G. Xu. Modeling with Cubic A-Patches. *ACM Transactions on Graphics*, 14(2):103–133, 1995.

[2] E. Catmull and J. Clark. Recursively Generated B-spline Surfaces on Arbitrary Topological Meshes. *Computer Aidded Design*, 10(6):350–355, 1978.

[3] L. Demkowicz and J. T. Oden. *Application of hp-Adaptive BE/FE Methods to Elastic Scattering*. Tech-

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 **IEEE**

Table 6.1: Errors and Computation Times

	Maximal Error -Real Part	Maximal Error -Imaginary part	Mean Error -Real Part	Mean Error -Imaginary part	Computation Time (second)
$Loop_{32}$	0.029642	0.275711	0.017419	0.109680	24
$Loop_{128}$	0.037112	0.053371	0.020714	0.022919	123
$Loop_{512}$	0.007949	0.007770	0.002676	0.002751	878
$Loop_{2048}$	0.001987	0.002846	0.000518	0.000561	9518
$Linear_{32}$	0.497239	2.053169	0.185843	0.598318	17
$Linear_{128}$	1.062628	1.459872	0.308296	0.451383	70
$Linear_{512}$	1.339224	1.035066	0.405807	0.414077	354
$Linear_{2048}$	1.406491	0.864905	0.443603	0.411614	3279

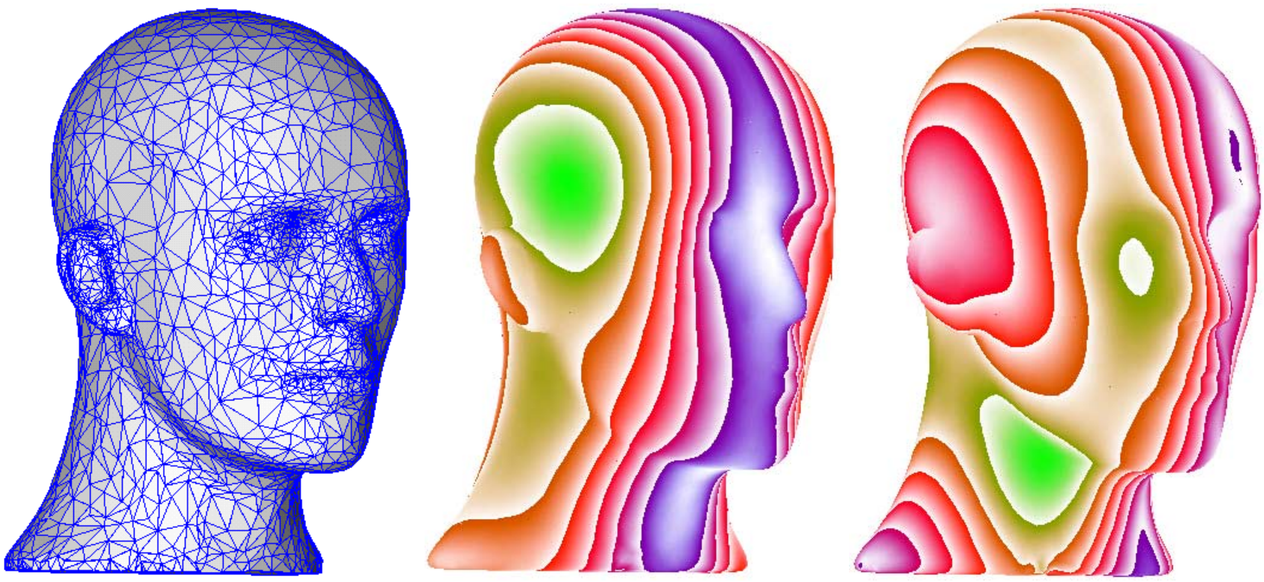


Fig 6.1: The geometric model of the human head used in our experiments (left). The iso-contour plots of the real and imaginary parts of the computed acoustics pressure function displayed on the human head surface (middle and right).

nical report, TICAM Report 94-15, The University of Texas At Austin, 1994.

[4] D. Doo and M. Sabin. Behavious of Recursive Division Surfaces near Extraordinary Points. *Computer Aidded Design*, 10(6):356–360, 1978.

[5] N. Dyn, D. Levin, and J. A. Gregory. A Butterfly Subdivision Scheme for Surface Interpolation with Tension Control. *ACM Transactions on Graphics*, 9(2):160–169, Apr. 1990.

[6] M. C. Junger and D. Feit. *Sounds, Structures, and their Interaction*. The MIT Press, 1972.

[7] L. Kobbelt, T. Hesse, H. Prautzsch, and K. Schweizerhof. Iterative Mesh Generation for FE-computation on Free Form Surfaces. *Engng. Comput.*, 14:806–820, 1997.

[8] C. T. Loop. *Smooth subdivision surfaces based on triangles. Master's thesis.* Technical report, Department of Mathematices, University of Utah, 1978.

[9] J. Peters and U. Reif. The simplest subdivision scheme for smoothing polyhedra. *ACM Transactions on Graphics*, 16(4):420–431, 1997.

[10] U. Reif. A unified approach to subdivision algorithms near extroordinary vertices. *Computer Aided Geometric Design*, 12:153–174, 1995.

[11] C. Schwab. Variable order composite quadrature of singular and nearly singular integrals. *Computing*, 53:173–194, 1994.

[12] J. Stam. Fast Evaluation of Catmull-Clark Subdivision Surfaces at Arbitrary Parameter Values. In *SIGGRAPH '98 Proceedings*, pages 395–404, 1998.

[13] J. Stam. Fast Evaluation of Loop Triangular Subdivision Surfaces at Arbitrary Parameter Values. In *SIGGRAPH '98 Proceedings, CD-ROM supplement*, 1998.

[14] T. F. Walsh. *HP Boundary Element Modeling of the Acoustical Transfer Properties of the Human Head/Ear* PhD thesis, Ticam, The University of Texas at Austin, 2000.

[15] J. Warren. Subdivision method for geometric design, 1995.

[16] D. Zorin, P. Schröder, and W. Sweldens. Subdivision for meshes with arbitrary topology. In *SIGGRAPH '96 Proceedings*, pages 71–78, 1996.

Proceedings of the Geometric Modeling and Processing — Theory and Applications
0-7695-1674-2/02 $17.00 © 2002 **IEEE**