Hierarchical Lattice Structures with Tetrahedral Symmetry

RUIQI CHEN, HARDIK KABARIA, and SAIGOPAL NELATURI, Carbon, Inc., USA
ADRIAN LEW, Stanford University, USA

TODO

1 INTRODUCTION

1.1 Contributions

2 RELATED WORKS

3 METHODOLOGY

3.1 Implicit Surfaces with Tetrahedral Symmetry

Tetrahedral symmetry, in the context of this work, refers to full or achiral tetrahedral symmetry, denoted as T_d. In other words, it is the symmetry group of a regular tetrahedron with uncolored faces. The symmetry group T_d consists of 24 symmetry operators. Each symmetry operator may be expressed as a 3×3 orthogonal matrix $\mathbf{Q}$ that represents a transformation of a point $\mathbf{x}$ about the origin. A geometric object possesses tetrahedral symmetry if applying any symmetry operator $\mathbf{Q} \in T_d$ to any arbitrary point on or inside the object results in a transformed point that remains on or inside the original object. A list of all 24 symmetry operators in T_d is given in Appendix A.1.

Implicit surfaces represent a geometric object through an equation of the form $\Phi(\mathbf{x}) = 0$ where $\Phi : \mathbb{R}^3 \to \mathbb{R}$ is some function that takes in a 3D query point $\mathbf{x}$ and returns zero if $\mathbf{x}$ is on the surface, returns a negative value if $\mathbf{x}$ is inside the surface, or returns a positive value if $\mathbf{x}$ is outside the surface (note that this convention is not universal and other authors may use the opposite convention, i.e. points inside the surface have positive return values). The magnitude of the return value may have some geometric meaning, but this is not required. For example, signed distance functions are a subset of implicit surface equations where the magnitude of the

Authors' addresses: Ruiqi Chen, rchensix@alumni.stanford.edu; Hardik Kabaria, hardik@carbon3d.com; Saigopal Nelaturi, snelaturi@carbon3d.com, Carbon, Inc., 1089 Mills Way, Redwood City, California, USA, 94063; Adrian Lew, lewa@stanford.edu, Stanford University, 440 Escondido Mall, Stanford, California, USA, 94305.

return value denotes the distance from the query point $\mathbf{x}$ to the surface.

To construct an implicit surface $\Phi(\mathbf{x})$ with tetrahedral symmetry, 3D basis functions that inherently possess tetrahedral symmetry are used. A 3D basis function with tetrahedral symmetry $\phi(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}$ may be constructed from some other conventional 3D basis function $\psi(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}$ by iterating over the symmetry operators $\mathbf{Q} \in T_d$, feeding the transformation $\mathbf{Qx}$ as input to ψ, and summing over all 24 possible values of $\mathbf{Q}$:

$$\phi(\mathbf{x}) = K \sum_{\mathbf{Q} \in T_d} \psi(\mathbf{Qx}) \tag{1}$$

where K is some normalization factor to make the basis function orthonormal. In this work, the 3D Fourier series

$$\psi_{\mathbf{f}}(\mathbf{x}) = e^{2\pi i \mathbf{f}^T \mathbf{x}} \tag{2}$$

is used, where $\mathbf{f} \in \mathcal{T}$ is an integer-valued vector that represents spatial frequencies in the Cartesian directions. The set of all possible parameters is given by $\mathcal{T}$:

$$\mathcal{T} = \left\{ \begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} \in \mathbb{Z}^3 | f_y + f_z \geq 0 \text{ and } f_y - f_z \geq 0 \text{ and } f_x - f_y \geq 0 \right\} \tag{3}$$

Note that the set $\mathcal{T}$ is a subset of $\mathbb{Z}^3$ to avoid duplicate basis functions, i.e. if $\mathcal{T}$ was all of $\mathbb{Z}^3$, then there will be multiple values of $\mathbf{f}$ that result in the same basis function $\phi_{\mathbf{f}}(\mathbf{x})$. Additionally, note that by writing the 3D Fourier series using complex exponentials (as expressed in Equation 2), Equation 1 is, in general, a function that takes in real-valued 3D points and returns complex valued scalars, i.e. $\phi_{\mathbf{f}}(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{C}$. However, for the purpose of generating implicit surfaces, only the real or imaginary part (but not both) of the final expression for $\Phi(\mathbf{x})$ will be used.

There are many reasons, both mathematical and practical, as to why the 3D Fourier series is an excellent choice for the foundational basis function, but the primary reasons are: 3D Fourier series are an orthonormal basis, 3D Fourier series have a very simple expression for the gradient (this is used in Section 3.3); there are fast, robust algorithms available (most notably, the fast Fourier transform) for computing the cofficients of a 3D Fourier expansion; and finally, several famous triply periodic minimal surfaces (TPMS) that are commonly used to construct hierarchical lattice structures (e.g. Schwarz primitive, gyroid) may be approximated using 3D Fourier series [Wohlgemuth et al. 2001].

Substituting Equation 2 into Equation 1, the basis function with tetrahedral symmetry may be expressed as

$$\phi_{\mathbf{f}}(\mathbf{x}) = K_{\mathbf{f}} \sum_{\mathbf{Q} \in T_d} e^{2\pi i \mathbf{f}^T \mathbf{Qx}} \tag{4}$$

where $K_{\mathbf{f}}$ is the normalization coefficient given by

$$K_{\mathbf{f}} = \frac{1}{\sqrt{\iiint_\Omega \left(\sum_{\mathbf{Q}_1, \mathbf{Q}_2 \in T_d} e^{2\pi i \mathbf{f}^T (\mathbf{Q}_1 - \mathbf{Q}_2) \mathbf{x}} \right) dV}} \quad (5)$$

where the region of integration Ω is the unit cube (with side length of 1) centered at the origin. In the integrand in the denominator of Equation 5, due to the orthonormal property of 3D Fourier series, only terms where $\mathbf{f}^T (\mathbf{Q}_1 - \mathbf{Q}_2) = \mathbf{0}$ contribute to the integral.

Just like the 3D Fourier series, the resulting basis function in Equation 4 is orthonormal, i.e.

$$\begin{aligned} \iiint_\Omega \phi_{\mathbf{f}}(\mathbf{x}) \overline{\phi_{\mathbf{g}}(\mathbf{x})} dV &= 0 \\ \iiint_\Omega \phi_{\mathbf{f}}(\mathbf{x}) \overline{\phi_{\mathbf{f}}(\mathbf{x})} dV &= 1 \end{aligned} \quad (6)$$

for some arbitrary $\mathbf{f} \in \mathcal{T}$, $\mathbf{g} \in \mathcal{T}$, and $\mathbf{f} \neq \mathbf{g}$. However, unlike the 3D Fourier series, Equation 4 also has tetrahedral symmetry. Finally, the implicit surface $\Phi(\mathbf{x})$ may be expressed in terms of the basis functions as

$$\Phi(\mathbf{x}) = \sum_{\mathbf{f} \in \mathcal{T}} c_{\mathbf{f}} \phi_{\mathbf{f}}(\mathbf{x}) \quad (7)$$

where $c_{\mathbf{f}}$ are the weight coefficients for each basis function. As previously mentioned, the return value of $\Phi(\mathbf{x})$ is, in general, complex-valued, so only the real or imaginary part of the result will be used.

Given an arbitrary input implicit surface $g(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}$, which may or may not possess tetrahedral symmetry, a tetrahedrally symmetric approximation to $g(\mathbf{x})$ may be computed by projecting $g(\mathbf{x})$ onto the basis functions to recover the weight coefficients:

$$c_{\mathbf{f}} = \iiint_\Omega g(\mathbf{x}) \phi_{\mathbf{f}}(\mathbf{x}) dV \quad (8)$$

In this case, because the input function $g(\mathbf{x})$ is real-valued, the output function $\Phi(\mathbf{x})$ will also be real-valued. In practice, Equation 8 will be evaluated numerically, and the input geometry to approximate does not have to be an implicit surface, as demonstrated in the following section.

3.2 Converting Geometry Representations to Implicit Surfaces

Frequently, a user will choose a unit cell geometry based on some criteria (e.g. aesthetics, porosity, mechanical performance, etc.) to tile to create the hierarchical lattice structure. The input unit cell to tile might not be an implicit surface; furthermore, it might not have tetrahedral symmetry either. To accomodate nearly arbitrary incoming geometry representations (e.g. triangle meshes, NURBS, voxel grids), the only requirement is that the input representation has a method to determine whether a query point is inside or outside the geometry. For example, inside/outside queries on oriented point clouds and triangle meshes/soups can be computed extremely robustly using the fast winding number method [Barill et al. 2018].

To numerically compute the implicit surface approximation $\Phi(\mathbf{x})$, first, the input geometry should be centered at the origin and be fully contained by the unit tetrahedron described in TODO. The input geometry is then sampled by querying the inside/outside status on an $N \times N \times N$ grid of query points. The query points

should equispaced sampled on the domain $[-0.5, 0.5]$, meaning the spacing between adjacement points in the Cartesian directions is $\frac{1}{N}$. Inside points are assigned a scalar value of -1 while outside points are assigned a scalar value of $+1$. Next, the fast Fourier transform (FFT) is applied to the inside/outside samples. Finally, the weight coefficients $c_{\mathbf{f}}$ may be computed from the FFT result by noting that the right hand side of Equation 5 is simply a summation of FFT coefficients due to the fact that $\phi_{\mathbf{f}}(\mathbf{x})$ is a summation of 3D complex exponentials.

It is worth noting that if the input geometry is band limited (i.e. the input geometry is generated from a 3D Fourier series implicit equation with a finite set of spatial frequencies), then using the FFT to compute the weight coefficients $c_{\mathbf{f}}$ will be exact up to limitations in floating point computations, as long as N is large enough to satisfy the Nyquist-Shannon sampling theorem, i.e. $h < \frac{N}{2}$ where $h \in \mathbb{Z}$ is the maximum magnitude integer frequency in any Cartesian direction of the input geometry. However, if the input geometry is not band limited (e.g. there are sharp edges and corners), then this method will only yield approximate values of $c_{\mathbf{f}}$ due to aliasing. In practice, given some arbitrary input geometry that may or may not be band limited, one can choose some reasonable starting value for N such that the FFT can compute relatively quickly (for example $N = 2^7$), compute the implicit surface approximation to the input, and use some stopping criteria to determine if N is large enough. One example of stopping criteria is the Hausdorff distance between the input geometry and output approximation. Furthermore, choosing powers of two for values of N is practical due to exploiting efficiencies in the FFT algorithm. The topic of convergence of the Fourier series is well studied in literature. Since the main purpose of this paper is to develop methods to generate hierarchical structures, i.e. the main purpose is *not* about the convergence of 3D Fourier series or the tetrahedrally symmetric basis function, most examples in this paper will use some arbitrarily chosen value for N.

The preceding section utilized inside/outside queries of the input geometry to develop the implicit surface approximation. More useful results can be achieved if the signed distance of the input representation can be computed at query points. This enables finding a tetrahedrally symmetric implicit equation approximation to the signed distance function (SDF) of the input geometry. For example, once the SDF approximation is computed, it is trivial to compute dilations and erosions (see Section 3.5). Using the sampled SDF as input will also yield a smoother approximation since the inputs are no longer binary. Because of this, most examples in this paper use the sampled SDF to compute the implicit surface approximation.

TODO

REFERENCES

Gavin Barill, Neil G. Dickson, Ryan Schmidt, David I. W. Levin, and Alec Jacobson. 2018. Fast winding numbers for soups and clouds. *ACM Transactions on Graphics* 37, 4 (July 2018), 1–12. https://doi.org/10.1145/3197517.3201337

Meinhard Wohlgemuth, Nataliya Yufa, James Hoffman, and Edwin L. Thomas. 2001. Triply periodic bicontinuous cubic microdomain morphologies by symmetries. *Macromolecules* 34, 17 (July 2001), 6083–6089. https://doi.org/10.1021/ma0019499