■ 연구보고

On Measuring the Quality of 3-D Triangulation

Joon-Young Park
Dept. of Industrial Engineering, Han Sung University

Abstract

A new criterion, the solid angle, is introduced to measure the quality of tetrahedral mesh. This criterion is compared with the existing Delaunay triangulation criterion. The properties of solid angles have been studied and are proposed for utilization in 3-D meshing algorithms. Furthermore, difficulties of developing a 3-D algorithm that provides a lower bound on the smallest angle have been discussed.

1. Introduction

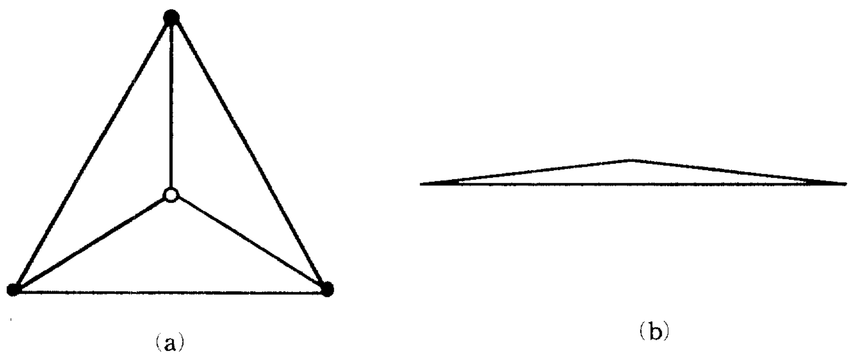
Finite element method is the most widely used scheme for the engineering analysis. In the finite element method, an important step is the mesh generation from the boundary description of the part. This divides the interior of the object into simple elements of a type known to the analysis program. Before a finite element analysis is performed, a finite element discretization of the problem must be generated. This discretization is generated for a given domain and a set of boundary conditions in a manner consistent with the differential equations used to describe the physical behavior being analyzed. In practice, quadrilaterals and triangles in 2-D, or bricks, wedges and tetrahedra in 3-D are used for discretization. Simplex elements (i.e., triangles in 2-D and tetrahedra in 3-D) are preferred because they accommodate complex boundaries of the object.

Since the mesh is a discrete approximation of the continuous interior of the object, it is important that the difference between mesh and object be small. Meshes were generated manually until the early 1960s. Since then, computers have been used to assist the task of discretizing complex objects. Programs for mesh generation have supported visualization and verification of the structure of nodes within elements and elements within meshes. Many algorithms have been developed to automate the mesh generation process [Ho-Le, 1988]. After the mesh generation, these algorithms use an iterative smoothing process to complete the pre-processing of the finite element analysis. In that process, elements with small angles are removed. Even with the smoothing process, most algorithms provides a guaranteed lower bound of the smallest angle in the mesh, and human intervention may sometimes be required to improve the mesh. Since the error in the finite element analysis is proportional to the smallest angle in a mesh [Strang, 1976], its lower bound guarantee is significant. With the same rationale, 3-D elements (tetrahedra) of the mesh need to be as close as possible to equilateral tetrahedra in order for mesh quality to be improved.

There have been a few attempts to improve the mesh quality by achieving equilateral triangulations as possible. Some algorithms achieve a guaranteed lower bound of the smallest angle of a mesh. For example, Rivara (1987) showed that the lower bound of the smallest angle in the mesh can be a half of the smallest angle in the initial triangulation. The triangulation is done by bisecting the longest edge of a triangle. To avoid the conformal problem, the process should be iteratively applied until conformity is satisfied. Field (1987) improved the mesh quality using the criterion of the Delaunay triangulation. Baker, Grosse, and Rafferty (1988) have presented a 2-D triangulation algorithm with a lower bound guarantee. Also, there are several 3-D triangular meshing algorithms. Many of these algorithms use Delaunay triangulation criterion. The reason for using this criterion is conjectured that the criterion generates a quality mesh in 2-D. However, according to a survey [Dey 1991], none of them generates a guaranteed quality mesh in 3-D. In this paper, a new criterion for measuring the quality of the tetrahedral mesh is introduced and a triangulation method using this criterion is investigated.

2. Measure of Tetrahedron

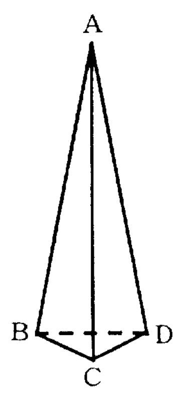
The most widely used criterion to measure the quality of mesh in 2-D is the face angle, the angle between two edges [Field, 1987: Rivara, 1987]. If the smallest face angle in a given mesh is increased, mesh quality is improved. The Delaunay triangulation in 2-D is proved to guarantee this criterion of maximizing the smallest angle [Joe, 1986]. Many algorithms in 3-D are also using Delaunay triagulation because it results a valid triangulation when a point set is given. However, in 3-D, Delaunay triangulation can generate flat tetrahedra even the point set is uniformly distributed. These flat tetrahedra are reported as 'slivers' and they are undesirable in most applications [Cavendish, 1985]. These elements exist because the Delaunay triangulation 3-D are formed in a way to increase a face angle which does not guarantee the improvement of the mesh quality as explained below.



〈 Figure 1 〉 Flat Tetrahedron with Large Face Angles and Small Dihedral Angles

In Figure 1, three vertices of a tetrahedron form an equilateral triangle and the fourth vertex lies very close to the centroid of the equilateral triangle. Figure 1 (a) shows a top view and Figure 1 (b) shows a side view of the tetrahedron. Every face angle in the above tetrahedron is bigger than 30 degrees but the tetrahedron is far from the equilateral tetrahedron. The flatness of the tetrahedron comes from the small ***dihedral angle*** which is the angle between two faces.

However, increasing the dihedral angle alone does not guarantee the improvement of the mesh quality. For example, in Figure 2, three vertices of a tetrahedron form an equilateral triangle and the fourth vertex lies far above the centroid of the equilateral triangle.

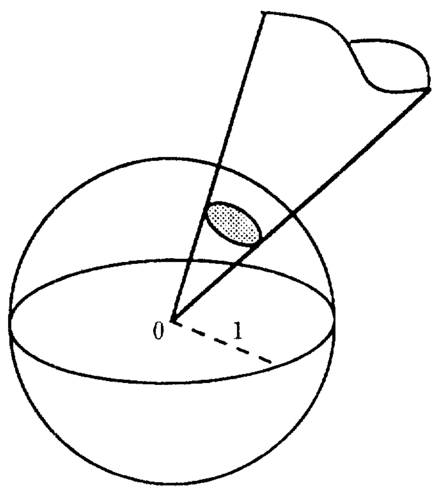


〈 Figure 2 〉 Sharp Tetrahedron with Small Face Angles and Large Dihedral Angles

The smallest dihedral angle in the above tetrahedron is close to 60 degrees (for example, dihedral angle between triangle ABC and ABD), but the tetrahedron is far from the equilateral tetrahedron. The sharpness of the tetrahedron comes from the small face angle (∠BAC). Therefore, face angles and dihedral angles should be increased at the same time to guarantee the improvement of the mesh quality.

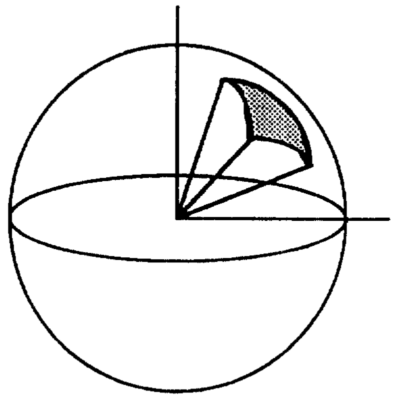
In this paper, therefore, the solid angle is introduced to measure the quality of the mesh in 3-D. The ***solid angle*** (a three dimensional angle) is defined as follows.

Definition 1 Solid Angle [Coxeter, 1965]
A solid angle is defined as a surface in space consisting of all half-lines which have a common initial point (the vertex), and which pass through a closed curve or a polygon (hence, a nappe of a conical surface). The solid angle *subtended at a point O* by a portion of a surface is the solid angle of all halflines from *O* which passes through the boundary of the portion.



⟨ Figure 3 ⟩ Measure of a Solid Angle

The notion of a solid angle in space generalizes the notion of an angle in the plane. The measure of a solid angle is the area it intercepts on a sphere of unit radius (when its vertex is at the center of the sphere); this is taken as the number of ***steradians*** in the angle, one ***steradian*** being a solid angle which intercepts a unit area on the unit sphere. For example, the solid angle of a right tetrahedron is $\pi/2$ steradians. A unit sphere has a total surface area of 4π which is the maximum value of the solid angle.



〈 Figure 4 〉 Solid Angle of a Tetrahedron

The following equation represents *the area of a triangle* (Δ) generated by intersecting a tetrahedron with a unit sphere as shown in Figure 4.

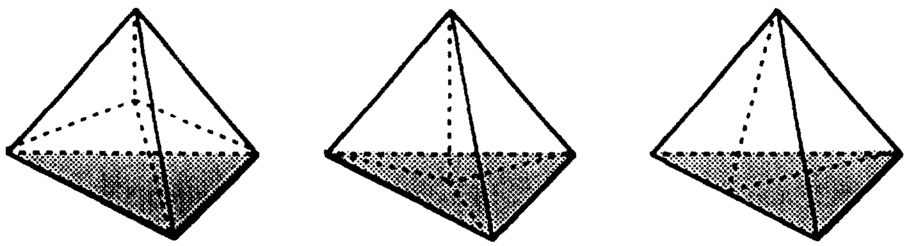
$$\Delta = 2(A + B + C - \pi) \qquad (1)$$

where, A, B, and C are the angles between a pair of great circles which pass through each edge of the triangle on the unit sphere.

3. Basic Operators in 3-D

In this section, a 3-D triangulation scheme is investigated using the solid angle criterion explained earlier. Although a complete algorithm is not provided in this paper, the investigation is expected to provide insights which lead to a 3-D triangulation algorithm. The basic approach of the triangulation scheme is to planting nodes first and connect these nodes to form tetrahedral elements.

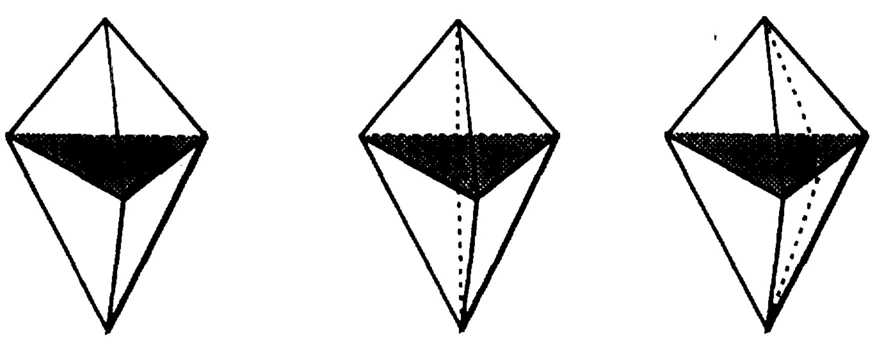
The planting of one node in a tetrahedron can be categorized into three operators (Figure 5) depending on the node location: **NI** (Node-in-Interior), **NF** (Node-on-Face), and **NE** (Node-on-Edge). **NI** divides one tetrahedron into four smaller tetrahedra. Every solid angle is divided into three smaller solid angles. **NI** does not have a conformity problem (Figure 5 [a]). **NF** divides one tetrahedron into three smaller tetrahedra. One solid angle is divided into three smaller solid angles, and each of the other three solid angles is divided into two smaller solid angles. Conformity should be met in one face (Figure 5 [b]). **NE** divides one tetrahedron into two smaller tetrahedra. Two solid angles are divided into two smaller solid angles each and the other two solid angles remain the same. Conformity should be met in two faces (Figure 5 [c]).



(a) **NI** (b) **NF** (c) **NE**

〈 Figure 5 〉 Three Operators in a Tetrahedron

When two tetrahedra are met, the resulting hexahedron is always convex if the shared triangle is the largest triangle (**LT**) in the region [Park, 1991]. Depending on the new node location in this convex region, several operators can be considered (Figure 6). When a new node is on **LT**, **NF** can be treated as a degenerate case of **NI**, since **LT** is inside of the convex region. Among these operators, it can be noted that **NE** is the most desirable operator in maximizing the smallest solid angle, since it divides an existing solid angle into at most two angles, and since it is the only operator which shortens the existing edge However, because of the conformity problem, **NI** and **NF** are also necessary.



(a) Hexahedron (b) **NI** or **NF** (c) **NE**

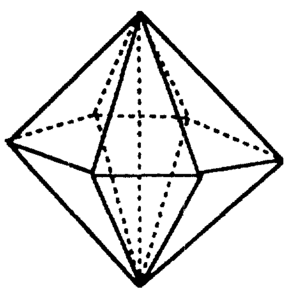
〈 Figure 6 〉 Three Operators in a Hexahedron

4. Edge Shortening Schemes

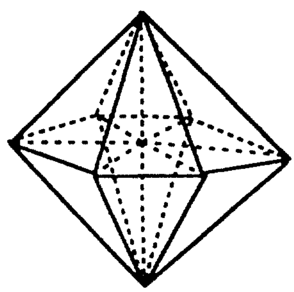
In order to decompose a given object into tetrahedra with a pre-defined element size, edge shortening schemes are necessary. Since **NI** and **NF** do not reduce the length of existing edges, the solid angles of elements will be degenerated if these operators are applied alone. Therefore, **NE** and **ES** (Edge Swapping) are discussed as edge shortening schemes.

4.2 Node Planting on Existing Edges (NE)

When **NE** is applied, all tetrahedra connected to the edge should be triangulated in order to satisfy conformity (Figure 7). **LE** in **LT** or **LE** in the mesh can be used to plant a new node. When **LE** in **LT** of the mesh is used, only two solid angles facing **LT** are guaranteed to be greater than the angle in an equilateral tetrahedron, and there is no guarantee that **LE** in the mesh is shortened. When **LE** in the mesh is used, solid angles facing the triangles, which share **LE**, are not necessarily bigger than an angle in an equilateral tetrahedron. **NE** reduces the element size by shortening **LE** length. However, since it does not enlarge small solid angles by changing the connectivity, this operator may result in degeneracies in the solid angle of the elements.



(a) **LE** Surrounding Tetrahedra

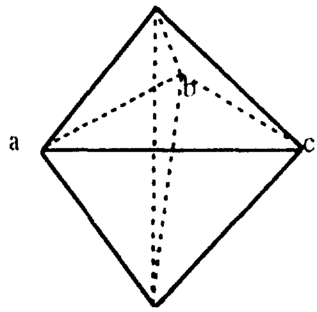


(b) **NE** on **LE**

〈 Figure 7 〉 Triangulation after **NE**

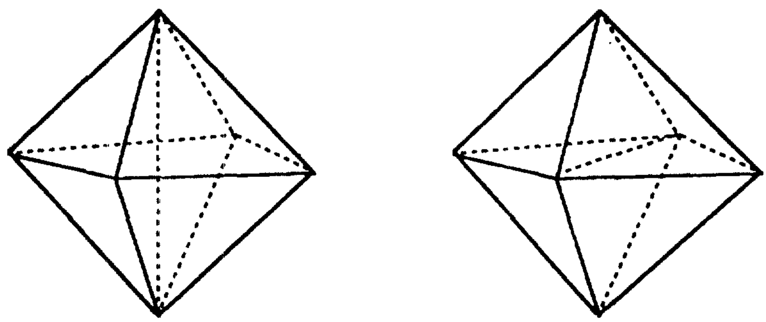
4.2 Edge Swapping Operator (ES)

In 2-D, **LE** defines a convex quadrilateral in which the swapping operation is always possible. In 3-D, however, **LE** defines an n-faceted polyhedral region which is not necessarily convex and the swapping operation is not always possible.



⟨ Figure 8⟩ Degenerate Case of Topological Operator

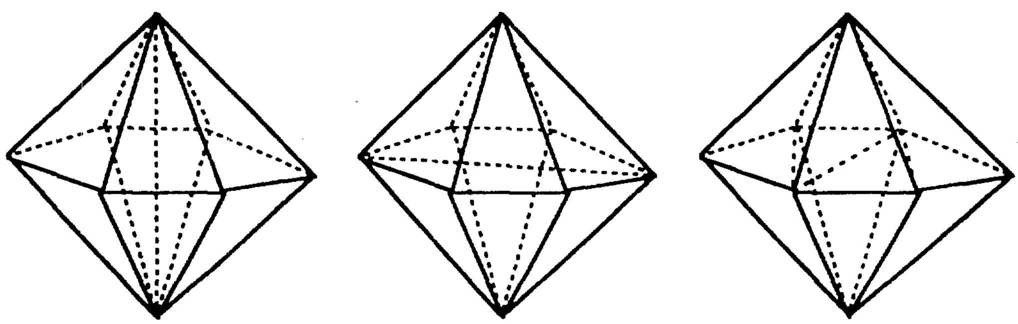
In Figure 8, **LE** is shared by three tetrahedra. Since the 3-D polygon *abc* is already simplex, the swapping is not possible. In this case, **LE** can be deleted satisfying a valid triangulation.



(a) **LE** Surrounding Tetrahedra (b) **ES** of **LE**

⟨ Figure 9 ⟩ Non-unique Swapping

In Figure 9, **LE** is shared by four tetrahedra. In this case, the swapping operation is not uniquely defined, since there are two ways of triangulating the quadrilateral surrounding the **LE** after the swapping operation.



⟨ Figure 10 ⟩ Triangulation with Multiple Edges after Swapping

In Figure 10, **LE** is shared by more than four tetrahedra. In this case, extra edges are necessary to complete the triangulation after the swapping operation. Unlike the **NE** operator, the **ES** operator changes the connectivity of the edges for the two end vertices of **LE**: hence enlarging the small solid angles. We can also notice that one extra edge creates four more solid angles.

Alternating between **NE** and **ES** should be determined carefully, since in 3-D more than one measure could be considered for node planting and triangulation. These measures are **LE** in the mesh, and **LE** in **LT**. Triangulation schemes also should be determined after the application of **ES**. The following table summarizes the effect of the **ES** operation, where n is the number of tetrahedra shared by **LE**.

〈 Table 1 〉 Changes of **ES** when $n \geqq 4$

	Before **ES**	After **ES**
Total # of Tetrahedra	n	$2n-4$
Max # of Solid Angles from one Vertex	n	$n-2$
Max # of Tetrahedra Shared by an Edge	n	4

From the Table 1, we can notice that **ES** creates more elements, while decreasing the maximum number of solid angles from one vertex and the maximum number of tetrahedra shared by an edge. Note that the maximum number of tetrahedra shared by an edge is drastically decreased from n to 4. Therefore, this operator improves mesh quality topologically. This topological improvement is important since it can lead to a geometrical improvement after applying smoothing schemes. However, if the goal is to guarantee a lower bound of the smallest solid angle, this geometrical improvement of the mesh quality has to be invested further.

5. Difficulty of 3-D Triangulation

There are several difficulties in 3-D triangulation compared to 2-D. For example, the number of elements shared by one edge is at most two in 2-D. However, the number of elements shared by one edge in 3-D could be up to $O(n)$, where n is the number of elements in the mesh. The following table summarizes the worst case adjacency information of 2-D and 3-D.

〈 Table 2 〉 Adjacency Information of 2-D and 3-D

	2-D	3-D
Vertex Adjacency	$O(n)$	$O(n)$
Edge Adjacency	2	$O(n)$
Face Adjacency	N/A	2

As we can see from the above table, there is an analogy between the edge adjacency in 2-D and the face adjacency in 3-D. Using this analogy, 3-D triangulation can be approached as the similar method to the 2-D triangulation of [Park 1991]. In order to avoid the degenerate case of creating solid angles with zero steradian, the swapping operator is necessary. However, since the swapping operator can not be defined by a face, edge swapping should take $O(n)$ computation time for each edge.

6. Conclusion

We investigated a solid angle as a new criterion for measuring the quality of 3-D mesh elements. This new criterion was compared with the existing Delaunay triangulation criterion. The properties of solid angles have been studied and are proposed for utilization in a 3-D meshing algorithm. Furthermore, difficulties of developing a 3-D algorithm that provides a lower bound on the smallest angle have been discussed.

A possible lower bound of the solid angle could be conjectured by the following process. One solid angle in an equilateral tetrahedron is 0.175π from equation (1). In the hexahedral convex region, defined by the merging of two tetrahedra sharing the largest triangle, **NF** and **NI** divide the largest solid angle into three smaller solid angles. From this observation, the lower bound in 3-D could be conjectured as $0.175\pi/3$. In this case the number of tetrahedra shared by one edge could be up to $O(n)$, where n is the total number of tetrahedra. Thus far, we do not have a way to handle n tetrahedra at the same time while satisfying the lower bound of the smallest solid angle. A detailed algorithm of 3-D remains as a future research task.

References

[1] Baker, B. S., Grosse, E. and Rafferty, C. S. (1988), "Non-obtuse Triangulation of Polygons," *Discrete and Computational Geometry*, Vol. 3, pp. 147-168.

[2] Cavendish, J. C., Field, D. A., and Frey, W. H. (1985), "An Approach to Automatic Three-Dimensional Finite Element Mesh Generation," *International Journal for Numerical Methods in Engineering*, Vol. 21, pp. 329-347.

[3] Coxeter, H. S. M. (1965), "Non-Euclidean Geometry," *University of Toronto Press*.

[4] Dey, T. K., Bajaj, C. L. and Sugihara, K. (1991), "On Good Triangulations in Three Dimensions," Submitted to ACM/SIGGRAPH Symposium on Solid Modeling Foundations and CAD/CAM Applications.

[5] Field, D. A. and Frey, W. H. (1985), "Automation of Tetrahedral Mesh Generation," *Research Publication GMR-4967*.

[6] Field, D. A. (1987), "A Flexible Delaunay Triangulation Algorithm," *Research Publication GMR-5675*.

[7] Ho-Le, K. (1988), "Finite Element Mesh Generation Methods: A Review and Classification," *Computer-aided Design*, Vol. 20, No. 1.

[8] Hoffmann, C. M. (1989), Geometric & Solid Modeling, Morgan Kaufmann Publishers, Inc.

[9] Joe, B. (1986), "Delaunay Triangular Meshes in Convex Polygons," *SIAM J. Sci. Com.*, Vol. 7, pp. 514-539.

[10] Kela, A. and Perucchio, R. (1986), "A Hierarchical Structure for Automatic Meshing and Adaptive FEM Analysis," Reliability of Methods for Engineering Analysis: Proceedings of the International Conference, University College, Swansea, 9th-11th, pp. 103-120.

[11] Kikuchi, N. (1986), Finite Element Methods in Mechanics, Cambridge.

[12] Lee, Y. T. (1984), "Automatic Finite-Element Mesh Generation from Geometric Models-A point-Based Approach," *ACM Transactions on Graphics*, Vol. 3, No. 4, pp. 287-311.

[13] Mantyla, M. (1988), *An Introduction to Solid Modeling*, *Computer Science Press*.

[14] Mortenson, M. E. (1985), *Geometric Modeling*, *John Wiley & Sons*.

[15] Park, J. Y. (1991), "Mesh Generation with Quasi-Equilateral Triangulation," *Ph. D Thesis*, Industrial and Operations Engineering Department, The University of Michigan, Ann Arbor, MI.

[16] Perucchio, R. Saxena, M., and Kela, A. (1989), "Automatic Mesh Generation from Solid Models Based on Recursive Spatial Decompositions," *International Journal for Numerical Methods in Engineering*, Vol. 28, pp. 2469-2501.

[17] Rivara, M. C. (1987), "A Grid Generator Based on 4-Triangles Conforming Mesh-Refinement Algorithms," *International Journal for Numerical Methods in Engineering*, Vol. 24, pp. 1343-1354.

[18] Ruppert, J. M. (1993), "Results on Triangulation and High Quality Mesh Generation," *Ph. D Thesis*, University of California, Berkely.

[19] Shephard, M. S. (1988), "Approaches to the Automatic Generation and Control of Finite Element Meshes," *Applied Mechanics Reviews*, Vol. 41, No. 4, pp. 169-185.

[20] Shephard, M. S., Yerry, M. A. and Baehman, P. L. (1986), "Automatic Mesh Generation Allowing for Efficient a Prior and Posteriori Mesh Refinement," *Computer Method in Applied Mechanics and Engineering*, Vol. 55, pp. 161-180.

[21] Strang, G. and Fix, G. (1976), *An Analysis of the Finite element Method*. *Prentice-Hall*

[22] Wordenweber, B. (1983), "Automatic Mesh Generation of 2 and 3 Dimension Curvilinear Manifolds," *Ph. D Thesis*, University of Cambridge.

[23] Yerry, M. A. (1985), "Automatic Mesh Generation for Three-Dimensional Solids," *Computers & Structures*, Vol. 20, No. 1-3, pp. 31-39.