Marching Cubes *33*: Construction of Topologically Correct Isosurfaces

Evgeni V. Chernyaev

Institute for High Energy Physics
142284, Protvino, Moscow Region,Russia
E-mail: chernaev@mx.ihep.su

Abstract

An algorithm implemented in the HIGZ graphics package for the construction of isosurfaces from volumetric datasets is discussed. This algorithm is an improved version of the Marching Cubes method. For each cell considered independently, the algorithm permits the construction of a triangle model, the topology of which coincides exactly with the topology of the isosurface of the trilinear function. It is shown that there are 33 topologically different configurations, instead of 15 as with the MC-method.

1 Introduction

The *Marching Cubes* (MC) method [1] is a well-known method for volume visualization. It produces triangle models of isosurfaces $F(x, y, z) = \alpha$ of a scalar function given by samples over a cuberille grid. Usually the MC-method is considered as the basic method for surface rendering in medical applications. However, it can also be applied in many other areas as, for example, for the visualization of implicitly specified functions or for the visualization of calculation results in the Finite Elements Method.

The MC-method processes one cell at a time. It determines how the isosurface intersects a given cell, and then moves to the next cell. Since there are eight nodes in each cube, and every node can be in two states, inside or outside the isosurface, there are $2^8 = 256$ different arrangements of a cube relative to the isosurface. However, most of the arrangements are topologically equivalent, and by rotation and/or by switching the states (in/out) of the nodes, they can be related to one of the 15 configurations shown in Figure 1.

These configurations are incorporated into a lookup table. Each entry in the table contains a triangle pattern for a corresponding configuration. For configuration 0 there are no triangles, because all eight nodes lie inside or outside the isosurface. For configuration 1 the pattern consists of one triangle, since the isosurface separates one node from the other seven. The patterns for other configurations have two, three or four triangles.

In addtion, the type of each cell in the grid is determined, and real vertices, formed by the intersection of the isosurface with the edges, are substituted in the corresponding triangle pattern. Linear interpolation along the edges is used to determine the coordinates of the vertices.

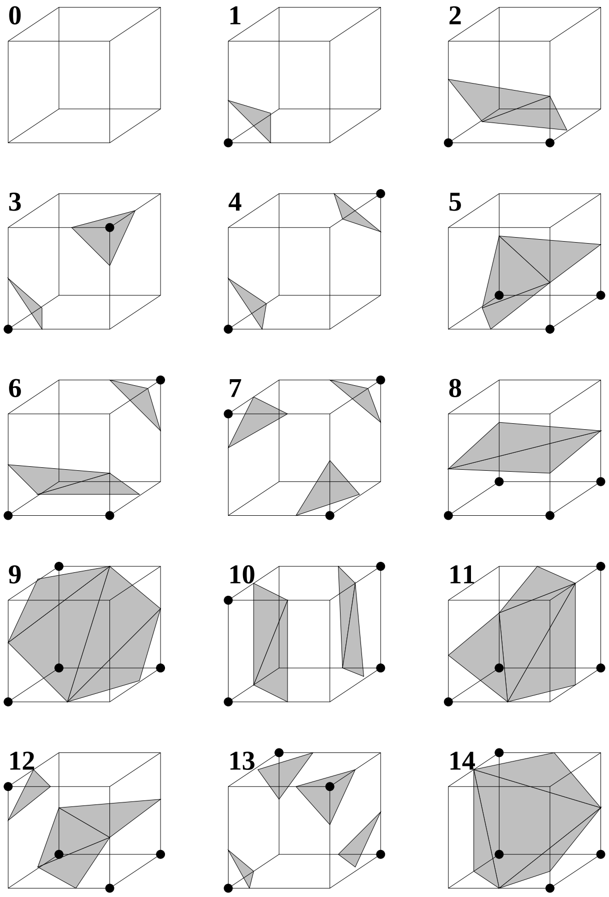

Figure 1: Original lookup table

The main problem with the MC-method is that there is the possibility for small "holes" to appear as a result of the discrepancy in the connection of the vertices on the shared face of two adjacent cells. Such a hole is shown in Figure 2, where a cube with configuration 6 shares a face with the complement of configuration 3. The shared face is the place where the hole appears.

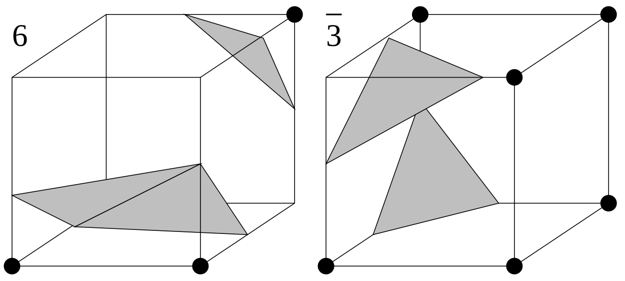

Figure 2: The hole on the shared face of two cells

Another problem is that even in the case where there are no visible discrepancies, the MC-method can produce isosurfaces with a topology different from that of isosurfaces produced for the same cells using other methods. Figure 3 shows isosurfaces produced for the same cude using two different methods. The first method is the MC-method, the second is the *Dividing Cubes* method [4], which subdivides the cube into smaller cubes that lie on the isosurface. The isosurface generated by the MC-method consists of two separate triangles, while the isosurface generated by the Dividing Cubes method looks like a "tube".

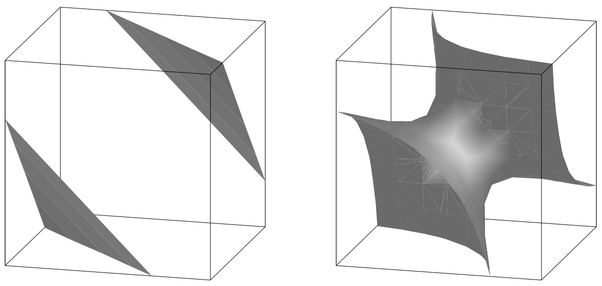

Figure 3: Isosurfaces generated by two different methods

The reason why the MC-method sometimes produces topologically incorrect results is explained as follows. In the case when each cell of a grid is processed independently of the surrounding cells, it is natural to use trilinear variation over the cube, as a direct extension to linear variation along an edge:

$$\begin{aligned} F(q,s,t) &= F_{000}\,(1-q)(1-s)(1-t) \\ &+ F_{100}\,q(1-s)(1-t) \\ &+ F_{010}\,(1-q)s(1-t) \\ &+ F_{110}\,qs(1-t) \\ &+ F_{001}\,(1-q)(1-s)t \\ &+ F_{101}\,q(1-s)t \\ &+ F_{011}\,(1-q)st \\ &+ F_{111}\,qst \end{aligned} \tag{1}$$

where q, s and t represent local coordinates of the cube, varying from 0 to 1, and $F_{000}, \ldots, F_{111}$ represent values at the corner nodes of the cube.

For each configuration in the lookup table the MC-method uses only one variant of the isosurface topology, while the trilinear function often permits several different variants. In this article we show that in order to take into account all the variants permitted by function (1), the number of entries in the lookup table should be increased from 15 to 33.

2 Terminology

First of all, let us formulate a concept of a **topologically correct model**. A triangle model, constructed for a cubic cell with a given set of values at the nodes, is **topologically correct** if its topology coincides with the topology of the trilinear function (1) specified for the given set of values.

Positive and negative nodes

As we have already seen, there are two types of node: some nodes lie outside the isosurface; others inside the isosurface. It is clear that the concepts **inside** and **outside** are conditional and can easily be reversed.

Usually, in papers on the MC-method, nodes of one type are designated as **marked** and nodes of the other type as **not marked**. We prefer to designate them as **positive** and **negative**. Although these designations are rather conditional, we can attribute a physical sense to them.

If the isovalue α is subtructed from all values over the grid, then the problem of the isosurface construction can be reformulated as a construction of isosurface $F(x, y, z) = 0$ on a grid with modified values at the nodes. For such a grid, nodes lying on different sides with respect to the isosurface, will have different signs.

Separated and non-separated nodes

We shall designate two nodes of the same sign as **non-separated** or **joined** if there is a path from one node to the other, along which the trilinear function (1) does not change sign. If such a path does not exist, we shall designate these nodes as **separated**.

It is important to know whether two nodes are separated or not. The required isosurface separates the positive and negative areas inside the cell, and the information about nodes allows us to determine the topology of the areas, and thus the topology of the isosurface.

Ambiguous face

In particular, the information about separated and non-separated nodes allows us to decide how to connect the vertices on a face. Since the function F varies linearly along the edges, it is obvious that two nodes located on the same edge are not separated. It is also obvious that if a face has three or four nodes of the same sign, then these nodes are non-separated. An ambiguity arises only in the case where a face has two positive and two negative diagonally opposed nodes. We call such a face an **ambiguous face**. All four edges of the ambiguous face are intersected by the isosurface. For the ambiguous face, the information on the node signs is insufficient to decide which nodes are separated and how to connect the vertices on the edges, and therefore it is necessary to carry out additional calculations.

Internal ambiguity

As we shall see later, in the majority of cases resolution of the ambiguities on the faces of a cell allows us to take a decision on the separability of all nodes of the cell. It is clear that in the case where for two nodes of the same sign there is a path along the faces from one node to the other, these nodes are non-separated. But if there is no such path along the faces, this does not mean that these two nodes are completely separated. The nodes could be joined inside the cell. In this case we shall say that the configuration has an **internal ambiguity**.

A typical example is configuration 4 (see Figure 4). There are no ambiguous faces for this configuration, but there can be two different cases: in one case the diagonal nodes are completely separated, in the other case they are joined inside the cell.

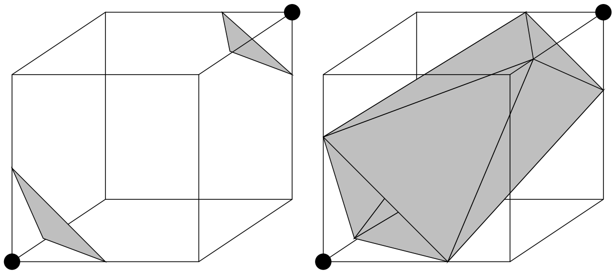

Figure 4: Configuration 4 – two possible cases

3 Resolving the ambiguities

In resolving the ambiguities on a face and inside a cube we shall use that the function F varies bilinearly over a face and over any plane parallel to a face. Indeed, if we fix any variable, for example we make $q = q_0$, then F takes the form:

$$\begin{aligned} F(s,t) &= A(1-s)(1-t) \;+\; Bs(1-t) \\ &+ \; C(1-s)t \;+\; Dst \end{aligned} \tag{2}$$

where

$$\begin{aligned} A &= F_{000}(1-q_0) \;+\; F_{100}(q_0) \\ B &= F_{010}(1-q_0) \;+\; F_{110}(q_0) \\ C &= F_{001}(1-q_0) \;+\; F_{101}(q_0) \\ D &= F_{011}(1-q_0) \;+\; F_{111}(q_0) \end{aligned}$$

Note that on a face, A, B, C, D are equal to the values at the corner nodes of the face.

Resolving the ambiguity on a face

A method of resolving the ambiguity on a face, based on the bilinear variation of F over the face, has been described by Nielson and Hamann [6]. Let A, B, C and D represent the values at the nodes of an ambiguous face, and let A, C be positive, while B, D are negative. It is easy to verify that the contour curve $F(s,t) = \alpha$ is a hyperbola. The decision concerning which nodes are separated and which are not, can be taken by comparing the isovalue α with the value $F(s_\alpha, t_\alpha)$ of the bilinear interpolant (2) at the point of intersection of the asymptotes (see Figure 5):

$$F(s_\alpha, t_\alpha) \;=\; \frac{AC - BD}{A + C - B - D} \tag{3}$$

Assuming that $\alpha = 0$, we can simplify the test. In order to determine which nodes are joined, it is sufficient to compare the two products AC and BD, because the denominator in expression (3) is always positive. So, if $AC > BD$, then the positive nodes are joined, and the negative nodes are separated; otherwise the positive nodes are separated, and the negative nodes are joined.

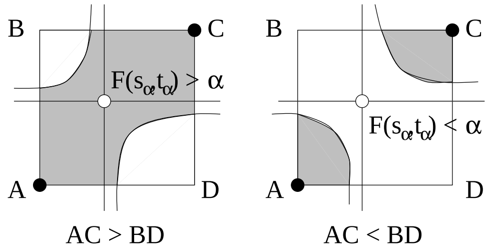

Figure 5: Resolving the ambiguity on a face

Resolving the internal ambiguity

There are different methods of resolving the internal ambiguity. One of them consists of the comparison of hyperbolas on the opposite faces of the cube where the internal ambiguity exists. If two areas of the same sign are joined inside the cube, then the projections of the hyperbolas must intersect each other. This follows from the linear behaviour of the trilinear function along a line parallel to an edge. In Figure 6 two different cases of configuration 4 are shown. In the first case the diagonal positive nodes are separated; in the second case these nodes are joined inside the cube. In the top views you can see that the projections of the hyperbolic arcs do not intersect each other in the first case, but do in the second case.

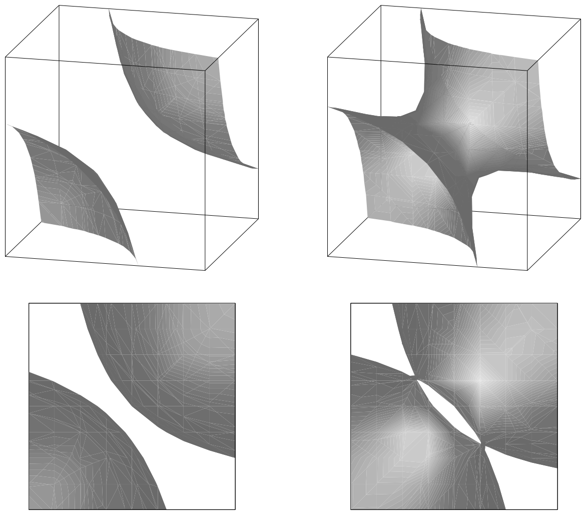

Figure 6: Two cases of configuration 4.

Another method is based on the bilinear variation of F over any plane parallel to a face. If inside a cube there are two areas of the same sign, which are separated on the faces but are joined inside the cube, then there exists a plane parallel to a face such that on the ambiguous face, formed by the intersection of this plane with the cube, the nodes of a given sign are joined (see Figure 7).

Let A_0, B_0, C_0, D_0 represent the values at the corner nodes on the face with $t = 0$, and A_1, B_1, C_1, D_1 represent the values at the corner nodes on the face with $t = 1$. Let the areas to be tested have a positive sign and adjoin the nodes with values A_0 and C_1. It is easy to verify that

$$A_0C_0 - B_0D_0 < 0$$
$$A_1C_1 - B_1D_1 < 0$$

If the areas are joined inside the cube, then there is t such that

$$A_tC_t - B_tD_t > 0 \tag{4}$$

Since function F varies linearly along the edges, we have

$$\begin{array}{rcl} A_t &=& A_0 \;+\; (A_1 - A_0)t \\ B_t &=& B_0 \;+\; (B_1 - B_0)t \\ C_t &=& C_0 \;+\; (C_1 - C_0)t \\ D_t &=& D_0 \;+\; (C_1 - C_0)t \end{array} \tag{5}$$

Substituting (5) into (4) we have a squared inequality

$$at^2 + bt + c > 0 \tag{6}$$

where

$$\begin{array}{rcl} a &=& (A_1 - A_0)(C_1 - C_0) \;-\; (B_1 - B_0)(D_1 - D_0) \\ b &=& C_0(A_1 - A_0) \;+\; A_0(C_1 - C_0) \\ &-& D_0(B_1 - B_0) \;-\; B_0(D_1 - D_0) \\ c &=& A_0C_0 \;-\; B_0D_0 \end{array}$$

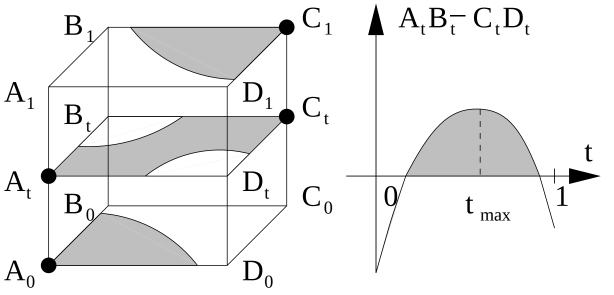

Figure 7: Resolving the internal ambiguity

It is easy to see, that the positive areas are joined only in the case when the parabola from the left side of (6) looks as shown in Figure 7: the branches are directed downwards; the maximum is positive and lies between 0 and 1. The algorithm for resolving the internal ambiguity consists of three steps:

- first, it is necessary to check that in (6), $a < 0$; if this is not true, then the positive areas are separated;
- next, it is necessary to check that $0 < t_{max} < 0$, where $t_{max} = -\frac{b}{2a}$; if this is not true, then the positive areas are separated;
- finally, it is necessary to check that the corner nodes of the face with $t = t_{max}$ have the required signs and that the inequality (4) is valid; if this is true, then the positive areas are joined; otherwise the positive areas are separated.

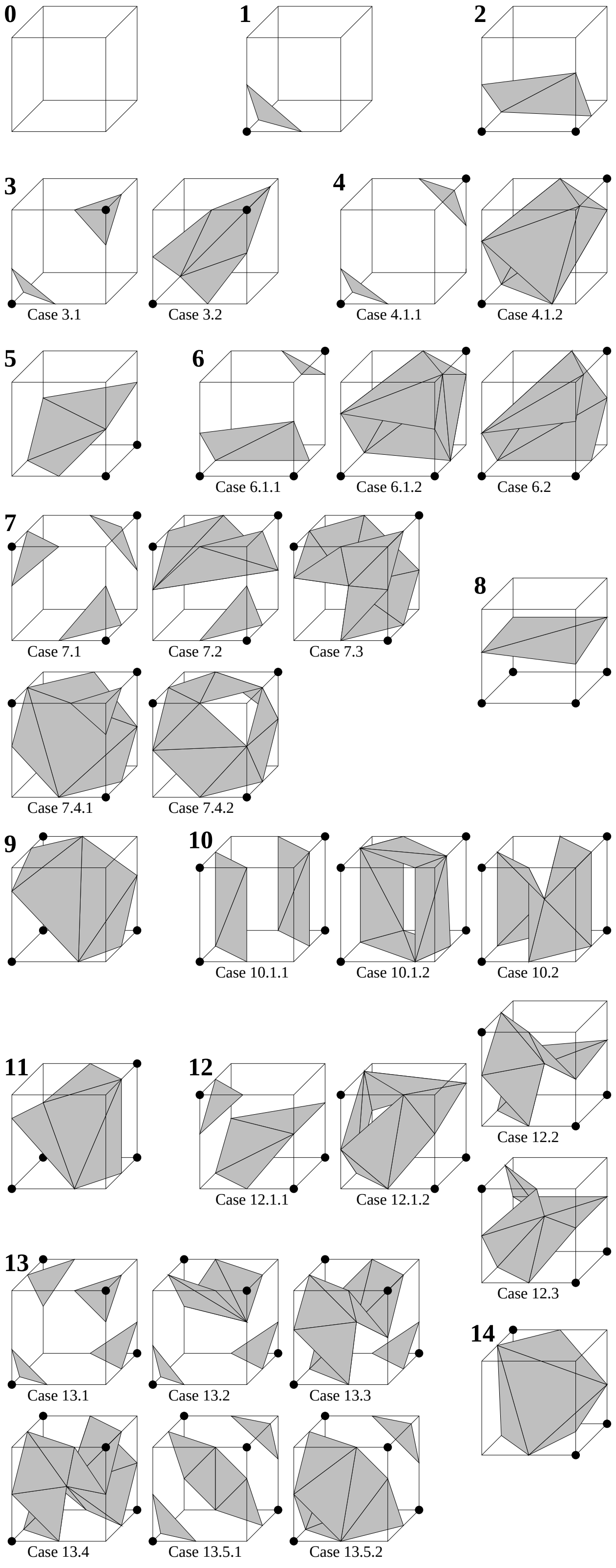

Figure 8: Advanced lookup table

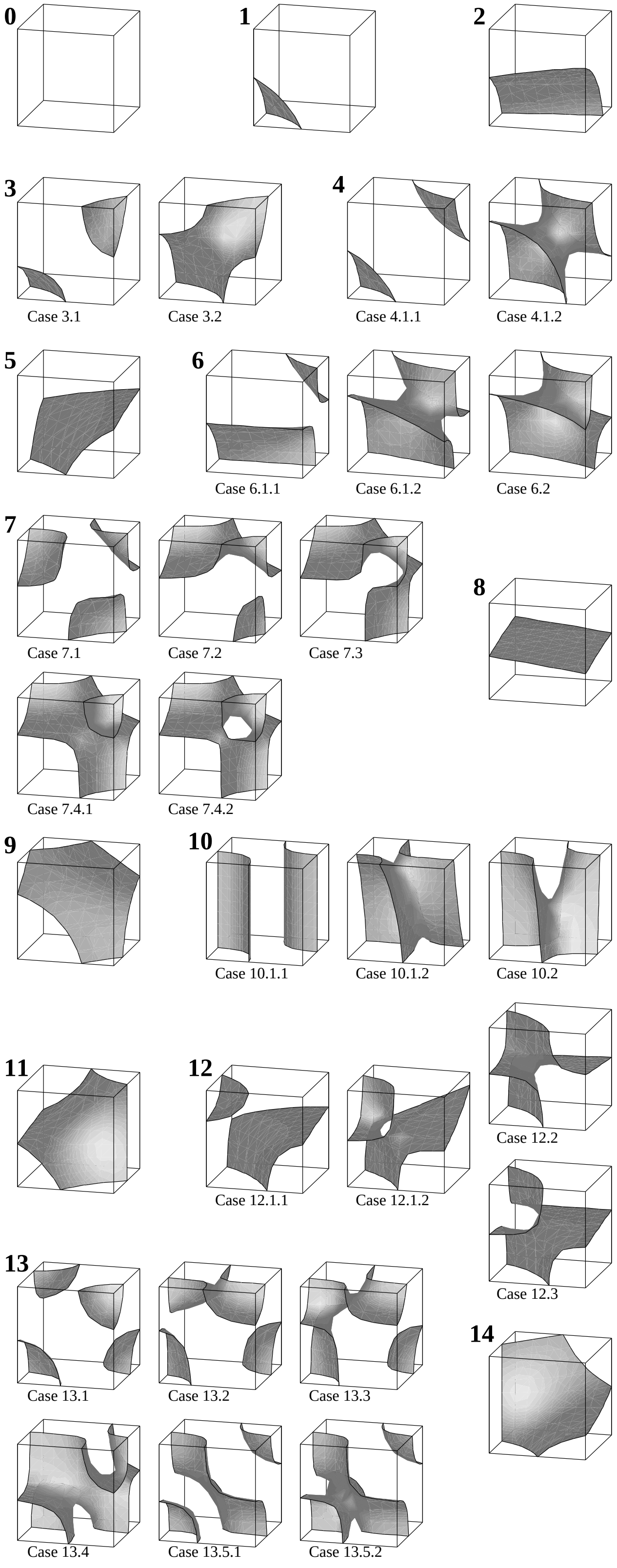

Figure 9: Real isosurfaces

4 Construction of topologically correct isosurfaces - various cases

Now that we have methods for resolving the ambiguities on a face and inside a cell, we can proceed to the consideration of various cases of the trilinear function behaviour inside the cube. All these cases are collected into an advanced lookup table shown in Figure 8. The corresponding isosurfaces are shown in Figure 9.

Simple configurations

Configurations 0, 1, 2, 5, 8, 9, 11 and 14 have no ambiguous faces and no internal ambiguity. All positive nodes, as well as all negative nodes, are joined. Note that configuration 14 is a mirror image of configuration 11.

Configuration 3

For this configuration there is only one ambiguous face. Resolving the ambiguity on this face results in two different cases:

case 3.1 – positive nodes are separated;

case 3.2 – positive nodes are not separated.

Configuration 4

For this configuration there are no ambiguous faces, but there is an internal ambiguity. Resolving the internal ambiguity results in two cases:

case 4.1.1 – positive diagonal nodes are completely separated;

case 4.1.2 – positive diagonal nodes are joined inside the cube.

Configuration 6

There is one ambiguous face. If positive nodes on this face are separated, then there is an internal ambiguity – these nodes can be either completely separated or can be joined inside the cube. Thus, three cases are possible:

case 6.1.1 – positive nodes of the ambiguous face are completely separated;

case 6.1.2 – positive nodes of the ambiguous face are joined inside the cube;

case 6.2 – positive nodes on the ambiguous face are not separated.

Configuration 7

There are three adjacent ambiguous faces for this configuration. Their common node is negative. If posi-tive nodes on all ambiguous faces are separated, then an internal ambiguity arises – their common negative node can be either completely separated from other negative nodes or joined with them inside the cube. Resolving the ambiguities results in five different cases:

case 7.1 – positive nodes are separated on all ambiguous faces;

case 7.2 – positive nodes are joined on one ambiguous face and are separated on two other ambiguous faces;

case 7.3 – positive nodes are joined on two ambiguous faces and are separated on the third ambiguous face;

case 7.4.1 – positive nodes are joined on all ambiguous faces; the negative node common to all ambiguous faces is completely separated from other negative nodes;

case 7.4.2 – positive nodes are joined on all ambiguous faces; the negative node common to all ambiguous faces, is joined inside the cube with other negative nodes.

In case 7.3, for the first time we have a situation when, for better conformity of the triangle model to the isosurface, it is necessary to use an additional vertex, situated inside the cube. It will be a common node for all triangles in the model (see Figure 10). One of the simplest ways to choose this vertex is to take it as the average of all other vertices.

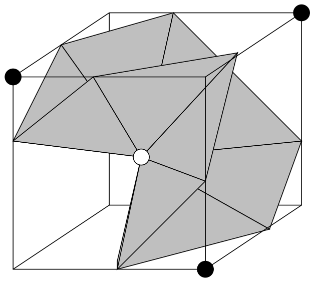

Figure 10: Case 7.3 - additional vertex

Configuration 10

In all previous configurations we have used the reverse of the node signs, so that the number of posi-tive nodes exceeded the number of negative nodes. We cannot use the reverse of signs for this purpose any more, since all remaining configurations have equal numbers of positive and negative nodes. However, by reversing the node signs we can ensure that one of the ambiguous faces will always have separated positive nodes. This allows us to reduce the number of different cases.

In particular, for configuration 10 we consider that the positive nodes on the top face are separated. So, the ambiguity exists only for the bottom face. If the positive nodes on this face are separated, then an internal ambiguity exists. Thus three different cases are possible:

case 10.1.1 – positive nodes on the top and bottom faces are separated; positive nodes located on the ends of large diagonals are also separated;

case 10.1.2 – positive nodes on the top and bottom faces are separated; positive nodes located on the ends of large diagonals are joined inside the cube;

case 10.2 – positive nodes on the top face are separated, but on the bottom face are not; in this case, as well as in case 7.3, an additional vertex is used for the construction of the triangle model.

Configuration 12

For configuration 12 there are two ambiguous faces. By reversing the signs it is always possible to ensure that positive nodes, at least on one of the ambiguous faces, are separated. In the case when positive nodes are separated on both ambiguous faces, then there is an internal ambiguity. In total, there are four different cases:

case 12.1.1 – positive nodes are separated on both ambiguous faces; their common positive node is completely separated from other positive nodes;

case 12.1.2 – positive nodes are separated on both ambiguous faces; their common positive node is joined inside the cube with other positive nodes;

cases 12.2 and 12.3 – these are mirror cases: positive nodes are separated on one ambiguous face and are not separated on the other; in both cases an additional vertex should be used for the construction of the triangle model.

Configuration 13

Configuration 13 is the most complex configuration. All six faces are ambiguous. Before beginning to consider the various cases, let us prove the following statement.

Statement 1 *The case in which positive nodes on two opposite faces are separated and on two other opposite faces are non-separated, is not possible for configuration 13.*

Proof: Let us assume that statement 1 is not valid, and that on the bottom and top faces of a cude, the positive nodes are joined, and on the front and rear faces they are separated. Let A_0, B_0, C_0, D_0 and A_1, B_1, C_1, D_1 represent the values at the corner nodes on the bottom and top faces respectively (see Figure 11). For the bottom and top faces we have:

$$\begin{array}{c} A_0C_0 > B_0D_0 \\ B_1D_1 > A_1C_1 \end{array} \tag{7}$$

On the front and rear faces the following inequalities are valid:

$$\begin{array}{c} A_0D_1 < A_1D_0 \\ B_1C_0 < B_0C_1 \end{array} \tag{8}$$

Note that all products in all inequalities are positive; therefore multiplication of the left and right parts in inequalities (7) and (8) produces a contradiction:

$$\begin{array}{c} A_0B_1C_0D_1 > A_1B_0C_1D_0 \\ A_0B_1C_0D_1 < A_1B_0C_1D_0 \end{array}$$

For the case of other pairs of opposite faces, we have the same contradiction. Statement 1 is proven.

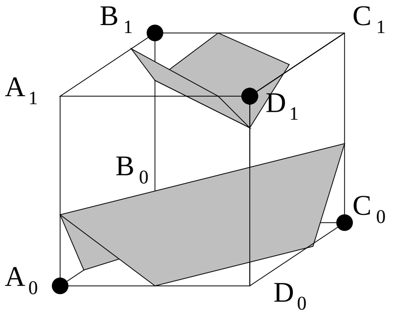

Figure 11: Case, not allowed for configuration 13

For configuration 13, by reversing the node signs it is always possible to ensure that the number of faces with separated positive nodes will be greater or equal to the number of faces where the positive nodes are joined. Therefore, it is necessary to consider only those cases where the number of faces with separated positive nodes is equal to 6, 5, 4 or 3. The first three cases are the following:

case 13.1 – positive nodes are separated on all faces;

case 13.2 – positive nodes are separated on five faces and are joined on one face;

case 13.3 – positive nodes are separated on four faces and are joined on two faces, which must be adjacent as follows from statement 1; for this case it is necessary to use an additional vertex to construct the triangle model.

Now it remains for us to consider the cases which have three faces with separated positive nodes. In accordance with statement 1 these faces must be adjacent. Their common node can be either positive or negative; in the case where it is positive, the internal ambiguity arises:

case 13.4 – the common node is negative; for construction of the triangle model it is necessary to use an additional vertex;

case 13.5.1 – the common node is positive and is completely separated from other positive nodes; opposite negative node is also completely separated from other negative nodes;

case 13.5.2 – the common node is positive and is joined inside the cube with other positive nodes; opposite negative node is again completely separated from other negative nodes.

Note that resolution of the internal ambiguity can result in a case where instead of the positive node, its opposite negative node will be joined inside the cube with other negative nodes. But this is not a different case, since by reversing the signs it can be transformed to case 13.5.2.

Thus, six different cases are possible for configuration 13, giving for all configurations 33 different cases.

5 Conclusions

In this article we have presented an advanced method for the construction of isosurfaces inside a cube. The concept of a topologically correct model has been formulated. It has been shown that there are two kinds of ambiguity: the ambiguity on a face and the internal ambiguity. Techniques for resolving both kinds of ambiguity have been considered. Finally, it has been shown that to cover all possible cases of topological behaviour of the trilinear function specified for a cube, the set of topologically different patterns should be increased from 15 to 33.

The described method was implemented in the framework of the PAW/HIGZ project [7], which is being carried out at the European Organization for Nuclear Research (CERN).

References

[1] W.E.Lorensen and H.E.Cline, "Marching Cubes: A High-Resolution 3D Surface Construction Algorithm", *Proceedings of SIGGRAPH '87; or Computer Graphics*, Vol. 21, No. 4, pp. 163-169, July 1987.

[2] G.Wyvill, C.McPheeters and B.Wyvill, "Data structures for soft objects", *The Visual Computer*, Vol. 2, No. 4, pp. 227-234, 1986.

[3] M.J.Dürst, "Additional reference to Marching Cubes", *Computer Graphics*, Vol. 22, No. 2, pp. 72-73, 1988.

[4] H.E. Cline, W.E. Lorensen, S. Ludke, C.R. Crawford and B.C. Teeter, "Two algorithms for the three-dimensional reconstruction of tomograms", *Medical Physics*, Vol. 15, No. 3, pp. 320-327, May/June 1988.

[5] J.Wilhelms and A.Van Gelder, "Topological Considerations in Isosurface Generation", *Computer Graphics*, Vol. 24, No. 5, pp. 79-86, 1990.

[6] G.M.Nielson and B.Hamann, "The Asymptotic Decider: Resolving the Ambiguity in Marching Cubes", *Proceedings of Visualization '91, IEEE Computer Society Press*, pp. 83-90, 1991.

[7] "PAW – Physics Analysis Workstation", *CERN Program Library Long Writeup Q121.*