Algorithmic aspects of alternating sum of volumes. Part 2: Nonconvergence and its remedy

K Tang and T Woo

The paper is the second part of a 2-part paper The first part focused on the issues of data structure and fast difference operation. The second studies the non-convergence of the alternating sum of volumes (ASV) process. An ASV is a series of convex components joined by alternating union and difference operations It is desirable that an ASV series be finite However, such is not always the case – the ASV algorithm can be nonconvergent The paper investigates the causes of this nonconvergence, and finds and proves the conditions responsible for it Linear time algorithms are then developed for detection

feature extraction, representation conversion, convex hull, alternating sum, difference operation, nonconvergence, supportability

An alternating sum of volumes (ASV) series is convergent if a deficiency $\boldsymbol{\Omega}_n$ is the null set; otherwise, it is said to be nonconvergent (For the computation of efficiency, the detection of a null deficiency $\boldsymbol{\Omega}_n$ can be replaced by the determination of the convexity of $\boldsymbol{\Omega}_{n-1}$) Figure 1 illustrates a nonconvergent ASV series. The series of deficiencies $\boldsymbol{\Omega}_1, \boldsymbol{\Omega}_2, \ldots,$ as derived from the convex hull (CH) and difference ($-$) operations, never converges to the null set, resulting in an infinite alternating series. $\{CH(\boldsymbol{\Omega}) - CH(\boldsymbol{\Omega}_1) + CH(\boldsymbol{\Omega}_2) - \cdots - CH(\boldsymbol{\Omega}_{2i-1}) + CH(\boldsymbol{\Omega}_{2i}) - \cdots\}$

As implied in Figure 1, the nonconvergence of an ASV series is determined by the nonconvergence of a deficiency in its expansion It is known[1] that an ASV series is nonconvergent when the convex hull of a deficiency $\boldsymbol{\Omega}_i$ is identified with the convex hull of the deficiency of $\boldsymbol{\Omega}_{i+1}$ For the example shown in Figure 1, the convex hull $CH(\boldsymbol{\Omega}_1)$ is equal to the convex hull $CH(\boldsymbol{\Omega}_2)$ As a result of the identification $CH(\boldsymbol{\Omega}_i) = CH(\boldsymbol{\Omega}_{i+1})$, the following relationship between the deficiencies holds $\boldsymbol{\Omega}_j = \boldsymbol{\Omega}_{j+2}$ $(i \leqslant j)$.

Formally, a deficiency $\boldsymbol{\Omega}_i$ is said to be *nonconvergent* if the convex hull of its deficiency $CH(\boldsymbol{\Omega}_i) - \boldsymbol{\Omega}_i$ is equal to $CH(\boldsymbol{\Omega}_i)$, and *convergent* otherwise It is desirable to be able to characterize the nonconvergence of a deficiency $\boldsymbol{\Omega}_i$ directly, rather than invoke the comparison between $CH(\boldsymbol{\Omega}_i)$ and $CH(\boldsymbol{\Omega}_i) - \boldsymbol{\Omega}_i$ This pursuit is justified in two respects. First, four convex-hull operations and two set-difference operations must be performed to obtain the datum $CH(\boldsymbol{\Omega}_i)$, $CH(\boldsymbol{\Omega}_i) - \boldsymbol{\Omega}_i$, and $CH(CH(\boldsymbol{\Omega}_i) - \boldsymbol{\Omega}_i)$ for the comparison. Set-difference operation on a polyhedron with m vertices is known to take at least $O(m^2)$ time prior to the $O(m\log m)$ result given in Part 1 of this paper[2]. Second, even if the fast $O(m\log m)$ difference operation is involved, detecting the presence of a null set, as the result of the difference, can be numerically unstable A fast nonconvergence detection algorithm for a

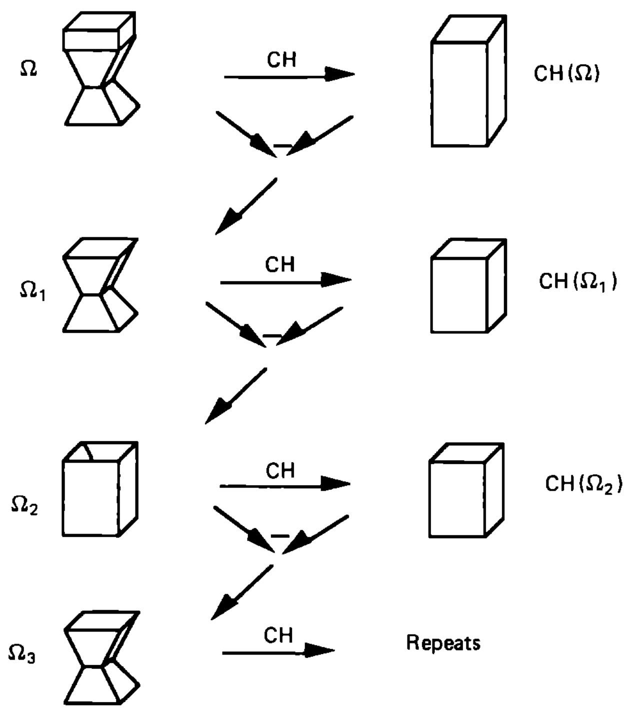

Figure 1 Illustration of ASV nonconvergence
[CH convex-hull operation, — difference operation]

Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI 48109-2117, USA
Paper received 13 June 1989 Revised 9 March 1990

pseudopolyhedron, where the set-difference and comparison operations are not carried out, is a new result in this Part 2 of the paper

Suppose that a fast nonconvergence detection algorithm for a deficiency is available One way in which to detect the nonconvergence of an ASV series is to test for the nonconvergence of every deficiency as it is being computed The time required by such a detection scheme is heavily dependent on the depth n of the first nonconvergent deficiency $\mathbf{\Omega}_n$ – the larger the number n, the more time it will take Alternatively, it may be asked whether the nonconvergence of a series can be detected without invoking the ASV process itself – not only because the deficiencies thus produced are nonproductive if the ASV series does not converge, but also because a separate scheme may speed up the detection time From the theoretical point of view, such a study addresses some interesting problems, such as that of finding the minimum number of faces in a nonconvergent deficiency

These two closely related issues, fast detection of the nonconvergence of a deficiency and that of an ASV series, are investigated in this paper In the next section, the concepts of *strong-hull* and *weak-hull vertices* are introduced The characterization of these two types of vertices leads to an $O(n\log n)$ algorithm for detecting the nonconvergence of a deficiency, where n is the number of vertices in the deficiency In the third section of the paper, a sufficient condition for the nonconvergence of an ASV series that requires only linear time to detect is given

CHARACTERIZATION OF NONCONVERGENT DEFICIENCIES

The following problem is solved in this section given a pseudopolyhedron $\mathbf{\Omega}_i$, under what condition will the equation $CH(\mathbf{\Omega}_i) = CH(CH(\mathbf{\Omega}_i) - \mathbf{\Omega}_i)$ hold, and how fast can such a condition be detected? The symbols CH and $-$ represent the convex hull and regularized difference operations, respectively (Note that every deficiency in an ASV series must be a pseudopolyhedron, as shown in Part 1 of this paper Hereafter, the two terms 'pseudopolyhedron' and 'deficiency' will be used interchangeably) Before the condition for nonconvergence is characterized, it is useful to summarize the relationships between the boundary and interior points of a pseudopolyhedron $\mathbf{\Omega}_i$, its convex hull $CH(\mathbf{\Omega}_i)$, and its deficiency $CH(\mathbf{\Omega}_i) - \mathbf{\Omega}_i$ The first relationship, given in Part 1 of this paper, is recited below

Lemma 1 The deficiency of a pseudopolyhedron $\mathbf{\Omega}_i$ is also a pseudopolyhedron, whose interior $\mathbf{I}(CH(\mathbf{\Omega}_i) - \mathbf{\Omega}_i)$ is the set difference $\{\mathbf{I}(CH(\mathbf{\Omega}_i)) - \mathbf{I}(\mathbf{\Omega}_i)\}$, and the boundary $\mathbf{B}(CH(\mathbf{\Omega}_i) - \mathbf{\Omega}_i)$ is a subset of $\{\mathbf{B}(CH(\mathbf{\Omega}_i)) - \mathbf{B}(\mathbf{\Omega}_i)\}$ that forms the closure of $\{\mathbf{I}(CH(\mathbf{\Omega}_i)) - \mathbf{I}(\mathbf{\Omega}_i)\}$

A pseudopolyhedron is completely described by its faces, and a face is determined by its edges, which are themselves defined by their end points, called vertices As the set of the vertices of the convex hull of a set of points must be a subset of that point set, by Lemma 1, the set of the vertices of the deficiency of $\mathbf{\Omega}_i$ is a subset of the vertices of $\mathbf{\Omega}_i$ In other words, the difference operation in the ASV expansion can be viewed as a *vertex-elimination process* after each difference operation, the deficiency $\mathbf{\Omega}_i$ possesses fewer vertices than does the deficiency $\mathbf{\Omega}_{i-1}$, this process continues until a convex pseudopolyhedron $\mathbf{\Omega}_n$ is reached whose deficiency $\mathbf{\Omega}_{n+1}$ is the null set

If the vertices in the deficiencies cannot be eliminated through the difference operation, the ASV series does not converge A vertex of a pseudopolyhedron $\mathbf{\Omega}_i$ is *eliminatable* if it does not exist in its deficiency $CH(\mathbf{\Omega}_i) - \mathbf{\Omega}_i$, otherwise, it is *noneliminatable* A formal definition of the nonconvergence of the pseudopolyhedron is then in order

Definition 1 A pseudopolyhedron $\mathbf{\Omega}_i$ is nonconvergent if all of its vertices are noneliminatable, otherwise, it is convergent

To characterize the eliminatability of vertices in $\mathbf{\Omega}_i$, the vertices are categorized into two groups, *hull vertices* and *internal vertices* The hull vertices are those that are on the boundary of $CH(\mathbf{\Omega}_i)$, whereas those vertices of $\mathbf{\Omega}_i$ that are not on the boundary of $CH(\mathbf{\Omega}_i)$ are internal. Each of the internal vertices has a 3D neighborhood that is strictly inside $CH(\mathbf{\Omega}_i)$ Further, this neighborhood contains a subset of $\{\mathbf{I}(CH(\mathbf{\Omega}_i)) - \mathbf{I}(\mathbf{\Omega}_i)\}$, as an internal vertex is also a boundary point of $\mathbf{\Omega}_i$ Therefore, by Lemma 1, all the internal vertices are noneliminatable To study the eliminatability of the hull vertices, they are further separated into *weak* and *strong* hull vertices

Definition 2 In E^3, the 3D Euclidean space, a hull vertex of $\mathbf{\Omega}_i$ is *weak* if it has a 3D neighborhood that contains points in $\{\mathbf{\Omega}_i \cup \{E^3 - CH(\mathbf{\Omega}_i)\}\}$ only, otherwise, it is called a *strong hull vertex*

As shown in Figure 2, after a difference operation, strong hull and internal vertices remain, whereas all the weak hull vertices are eliminated Let those faces (edges) of a pseudopolyhedron $\mathbf{\Omega}_i$ be called *hull faces* (*hull edges*) if they are completely on the boundary surface of $CH(\mathbf{\Omega}_i)$, and *internal faces* (*internal edges*) otherwise Referring to Figure 2, it can be inferred that a hull vertex is weak if and only if all of its incident faces are hull faces of $\mathbf{\Omega}_i$ (Note, however, that this condition does not hold for incident edges That is, a hull vertex with incident hull edges only is not necessarily weak, as shown in Figure 3, where the strong hull vertex v has no incident internal edges) The contribution of strong hull vertices to the nonconvergence is demonstrated by the following lemma

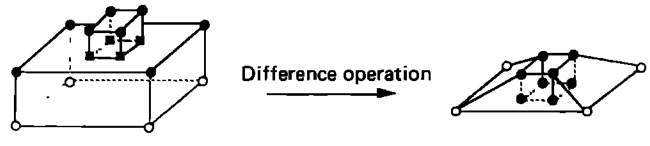

Figure 2. Weak hull vertices, strong hull vertices and internal vertices

[○ weak hull vertices, ● strong hull vertices, × internal vertices]

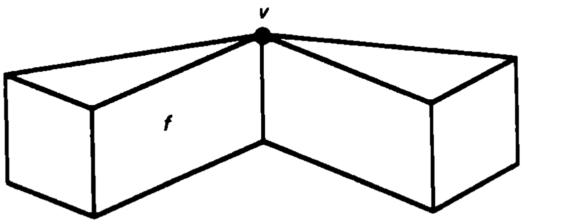

Figure 3 Strong hull vertex with no incident internal edges

Lemma 2. A pseudopolyhedron $\mathbf{\Omega}_i$ is nonconvergent if and only if all of its hull vertices are strong.

Proof. First, it is noted that the hull and internal vertices partition the entire vertex set of $\mathbf{\Omega}_i$, owing to their mutual exclusivity By Definition 2, a weak hull vertex has an open 3D neighborhood within which $\mathbf{\Omega}_i$ is equal to CH($\mathbf{\Omega}_i$), and thus there is no subset of $\{\mathbf{I}(\mathrm{CH}(\mathbf{\Omega}_i)) - \mathbf{I}(\mathbf{\Omega}_i)\}$ in that neighborhood Hence, by Lemma 1, all the weak hull vertices are eliminatable Conversely, as every 3D neighborhood of a strong hull vertex contains a subset of $\{\mathbf{I}(\mathrm{CH}(\mathbf{\Omega}_i)) - \mathbf{I}(\mathbf{\Omega}_i)\}$, these are preserved on the deficiency of $\mathbf{\Omega}_i$, i.e they are noneliminatable By Definition 1 and the fact that all the internal vertices are noneliminatable, the proof is complete QED

Lemma 2 implies that the detection of the nonconvergence of a pseudopolyhedron $\mathbf{\Omega}_i$ is equivalent to distinguishing its strong hull vertices from the weak ones Such a process requires two steps classify the hull and internal faces of $\mathbf{\Omega}_i$, and then check if $\mathbf{\Omega}_i$ has a vertex that has incident hull faces only. Whether a face is internal can be identified by checking one of its interior points. (Such a point must not be on an edge of the face, as an internal face may have hull edges only, e.g. face *f* in Figure 3.) $\mathbf{\Omega}_i$ is then nonconvergent if and only if no weak hull vertex exists.

The algorithm given below follows the two steps just described It is assumed that a procedure HULL(N, V, V_{tag}) is in hand that takes a list V of N points as input, and outputs a property array V_{tag} such that, if $V_{tag}(i)$ is 'true', point i in V is a hull vertex of CH(V), and if it is 'false', an internal vertex.

DETECT algorithm

```
Algorithm DETECT (Ω_i)
/* Detect the nonconvergence of a pseudopolyhedron Ω_i
The vertex list V and face list F of Ω_i have n_v vertices and n_f faces, respectively */

begin
(1)    for k = 1 to n_f do
           V(n_v + k) ← an interior point of face k in F
       end do
(2)    call HULL(n_v + n_f, V, V_tag)
(3)    set array VP(1 n_v) to 'true'
(4)    for k = 1 to n_f do
           for every vertex v of face k in F do
               VP(v) ← VP(v) ∩ V(n_v + k)
           end do
       end do
(5)    for k = 1 to n_v do
           if VP(k) = 'true' then
               return ('convergence )
           end if
       end do
(6)    return ('nonconvergence')
end DETECT
```

In the DETECT algorithm, the n_f interior points of the faces of $\mathbf{\Omega}_i$ are first appended to the vertex array V of $\mathbf{\Omega}_i$. As each interior point of a face can be obtained in constant time by considering any two adjacent edges of that face, Step 1 takes $O(n_f)$ time. The convex-hull procedure HULL is called at Step 2, which requires only $O((n_v + n_f)\log(n_v + n_f))$ time[3] At Step 3, a property array $VP(1:n_v)$ is preset to 'true'. At Step 4, the following is carried out if a face k is internal, i.e. its interior point tag $V_{tag}(n_v + k)$ is 'false', the corresponding entries in VP for all the vertices of face k are reset to 'false' Such a process obviously takes $O(D)$ time, where $D = \Sigma d_i$ ($i = 1, 2, \ldots, n_v$), and d_i is the degree of vertex i It is shown in the Appendix of Part 1 of this paper that D is $O(n_f)$. Finally, at Step 5, the array VP is scanned, and $\mathbf{\Omega}_i$ is identified as convergent if some entry in VP is 'true', and as nonconvergent otherwise. The time complexity of the DETECT algorithm is summarized by the following theorem.

Theorem 1 The detection of the nonconvergence of a pseudopolyhedron $\mathbf{\Omega}_i$ with n vertices can be done in $O(n\log n)$ time.

Compared with the simple comparison method[1], $\mathrm{CH}(\mathbf{\Omega}_i) = \mathrm{CH}(\mathrm{CH}(\mathbf{\Omega}_i) - \mathbf{\Omega}_i)$, the new detection algorithm DETECT avoids both the time-consuming difference operation and the identification of a null set that could be numerically unstable Two convex-hull operations are also saved

It may be noted that the detection algorithm DETECT disregards the disconnectedness of a set The pseudopolyhedron $\mathbf{\Omega}_i$ in Figure 4(a) is nonconvergent, by Lemma 2 The deficiency $\mathbf{\Omega}_{i+1}$, however, consists of two separate pseudopolyhedra $\mathbf{P}_1$ and $\mathbf{P}_2$. Although $\mathbf{\Omega}_{i+1}$ is nonconvergent as a single set, it is convergent if represented as $\mathrm{ASV}(\mathbf{\Omega}_{i+1}) = \mathrm{ASV}(\mathbf{P}_1 + \mathbf{P}_2) = \mathrm{ASV}(\mathbf{P}_1) + \mathrm{ASV}(\mathbf{P}_2)$, because $\mathbf{P}_1$ and $\mathbf{P}_2$ are both convergent It results in a convergent ASV tree $\mathrm{ASV}(\mathbf{\Omega}_i) = \mathbf{H} - \mathbf{\Omega}_{i+1} = \mathbf{H} - \mathrm{ASV}(\mathbf{\Omega}_{i+1}) = \mathbf{H} - (\mathrm{ASV}(\mathbf{P}_1) + \mathrm{ASV}(\mathbf{P}_2))$, which branches at the deficiency $\mathbf{\Omega}_{i+1}$ In some other cases, a pseudopolyhedron, though connected, might be separated uniquely at some edges such that the separated subsets

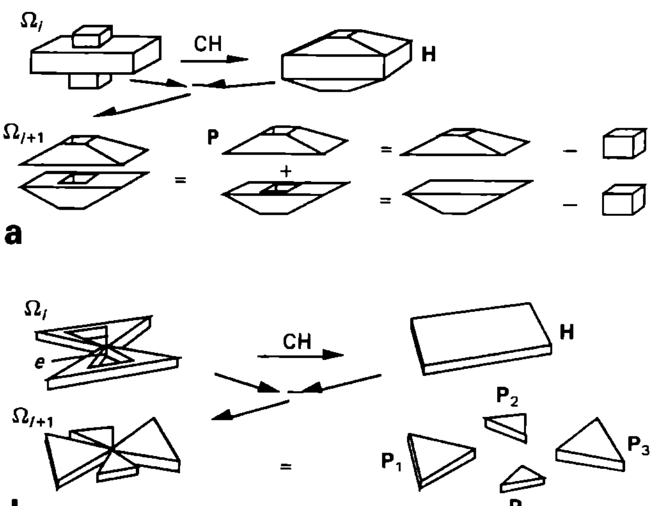

Figure 4 Convergence by set separation

Figure 5 Well connected set versus ill connected set, (a) well connected set, (b) ill connected set

are all convergent. For example, the pseudopolyhedron $\mathbf{\Omega}_i$ in Figure 4(b) is nonconvergent, by Lemma 2. It is, however, convergent if expressed as $\mathbf{\Omega}_i = \mathbf{H} - \mathbf{\Omega}_{i+1} = \mathbf{H} - (\mathbf{P}_1 + \mathbf{P}_2 + \mathbf{P}_3 + \mathbf{P}_4)$, as $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$ and $\mathbf{P}_4$ are all convergent.

Both examples of set separation on $\mathbf{\Omega}_{i+1}$ shown in Figure 4 have a crucial property: the boundary of the separated pseudopolyhedron remains unchanged. Unlike in polyhedral decomposition[4], such a property guarantees that the boundary after the separation has the same sets of vertices, edges and faces as before, with only the adjacency and incidence relationships between them being altered. Further, it is next shown that such a separation is unique, thus justifying the existence of a deterministic algorithm. To define set separation rigorously, the concept of *well connectedness* is needed.

Definition 3 Two points p and q of a pseudopolyhedron $\mathbf{\Omega}_i$ are said to be *well connected* in $\mathbf{\Omega}_i$ if there exists a curve c between p and q such that all the points in c, except for possibly p and q, are in $\mathbf{I}(\mathbf{\Omega}_i)$. $\mathbf{\Omega}_i$ is a *well connected set* if all of its points are well connected in it, otherwise it is an *ill connected* set (see Figure 5).

The $\mathbf{\Omega}_{i+1}$ in Figure 4(a), and both $\mathbf{\Omega}_i$ and $\mathbf{\Omega}_{i+1}$ in Figure 4(b), are ill connected pseudopolyhedra. A well connected pseudopolyhedron is also called a *robust set*, meaning that its interior is all connected. A subset ζ of a pseudopolyhedron $\mathbf{\Omega}_i$ is a *maximally well connected set* (MWCS) of $\mathbf{\Omega}_i$ if ζ is a well connected set, and any addition of non-ζ points of $\mathbf{\Omega}_i$ to ζ will constitute an ill connected set. As an example, only $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$ and $\mathbf{P}_4$ are the MWCSs of the pseudopolyhedron $\mathbf{\Omega}_{i+1}$ in Figure 4(b).

It is desirable that an ASV series be expanded as much as possible so that more features can be extracted. Once a nonconvergent and ill connected deficiency is encountered, it should be separated into the MWCSs, and the ASV process can then be performed on each of these. This leads to the notion of *strong* and *weak* nonconvergence.

Definition 4 A nonconvergent pseudopolyhedron $\mathbf{\Omega}_i$ is *strongly nonconvergent* if both itself and its deficiency are robust. Otherwise, $\mathbf{\Omega}_i$ is *weakly nonconvergent*.

As examples, the deficiency $\mathbf{\Omega}_1$ in Figure 1 is strongly nonconvergent, as both it and its deficiency $\mathbf{\Omega}_2$ are robust, whereas each $\mathbf{\Omega}_i$ in Figure 4 is weakly nonconvergent, because either it, or its deficiency $\mathbf{\Omega}_{i+1}$, is ill connected.

The detection of the strength of nonconvergence of a pseudopolyhedron $\mathbf{\Omega}_i$ of n faces requires three steps: the identification of its nonconvergence, the computation for the deficiency of $\mathbf{\Omega}_i$, and the classification of the well connectedness of $\mathbf{\Omega}_i$ and/or its deficiency. The first step can be done, by Theorem 1, in $O(n\log n)$ time. The difference operation also requires $O(n\log n)$ time, as shown in Part 1 of this paper. Thus, if the classification of the robustness of $\mathbf{\Omega}_i$ can be done in $O(n\log n)$ time, then so can the detection of the strong nonconvergence. As $\mathbf{\Omega}_i$ is robust if and only if its MWCS separation contains only one MWCS, i.e. itself, the goal becomes the finding of an $O(n\log n)$ time MWCS separation algorithm.

In implementing such an algorithm, it is noted that, by the definition of a pseudopolyhedron, *the well connectedness of its boundary-point set will ensure that it is a well connected set*. This fact ensures that the MWCSs of a pseudopolyhedron can be detected by checking only the well connectedness of its faces.

Let two faces of a pseudopolyhedron be *well adjacent* to each other if they share a common edge and are well connected to each other. It can easily be shown that two faces A and B of a pseudopolyhedron are well connected if and only if either they are well adjacent, or there exists a number of faces $f_1, f_2, \ldots, f_k$ such that A is well adjacent to f_1, f_1 is well adjacent to f_2, ..., and f_k is well adjacent to B. For example, in Figure 6, faces A and B are not well connected, because the curve c connecting points p and q passes through edge e, which does not belong to the interior of that pseudopolyhedron.

To characterize face well adjacency, let $f_1, f_2, \ldots, f_m$ be the faces incident to a common edge, ordered by their spatial angles (In Figure 7, $f_1, f_2, \ldots, f_m$ are the intersections between the faces sharing a common

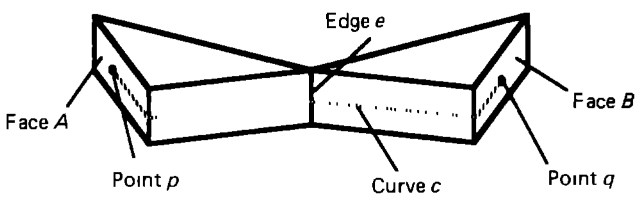

Figure 6 Ill connectedness of faces of a pseudopolyhedron

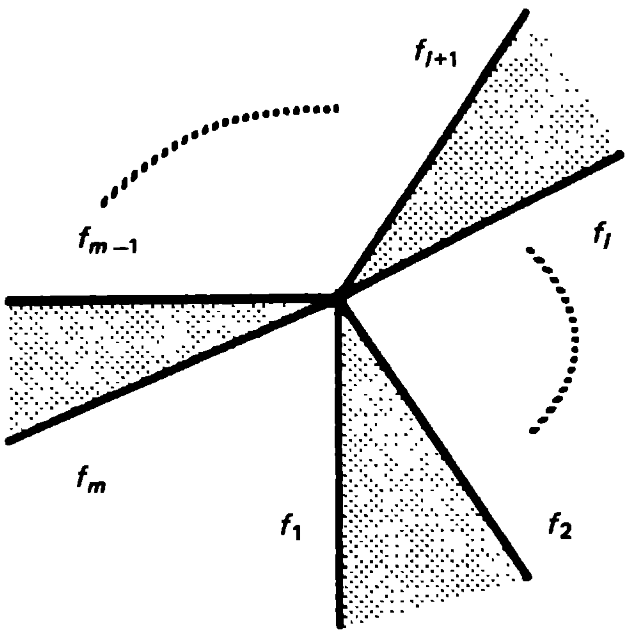

Figure 7 Well adjacency of faces

edge and a plane orthogonal to that edge.) Apparently, the well adjacent face of face f_i ($i = 1, 2, \ldots, m$) is either f_{i-1} or f_{i+1} (mod m), depending on the direction of the outward normal of f_i. Such a pairing process can be done in $O(m)$ time. The following recursive procedure MWCS_FACES finds the faces of a maximally well connected set of a pseudopolyhedron $\mathbf{\Omega}_i$. The input is the pseudopolyhedron representation $\{V, E, F, \text{NORM}, E_f\}$ of $\mathbf{\Omega}_i$ and the index of a face of $\mathbf{\Omega}_i$. The output is the indices of those faces of $\mathbf{\Omega}_i$ that form the boundary of an MWCS of $\mathbf{\Omega}_i$.

MWCS_FACES procedure

```
Procedure MWCS_FACES (f, Ω_i)
/*      Find those faces of a maximum well connected set of pseudopolyhedron Ω_i
        f is the index of a face that is required to be on the MWCS
*/
begin
(1)     output f
(2)     e_1, e_2, ..., e_k ← edges of face f
(3)     for j = 1, k do
(3.1)       f' ← the index of the face well adjacent to f at edge e_j
(3.2)       if (f' has not been output) then call MWCS_FACES(f', Ω_i)
        end do {Step 3}
end MWCS_FACES
```

Suppose that a total of m edges $e_1, e_2, \ldots, e_m$ of $\mathbf{\Omega}_i$ are found in an MWCS, denoted by P, through MWCS_FACES. Let k_i and k_i' be the face-adjacency indices of $\mathbf{\Omega}_i$ and P at edge e_i, respectively ($i = 1, 2, \ldots, m$). Step 1 takes constant time, and thus the overall time spent at Step 1 when MWCS_FACES terminates is $O(n)$, where n is the number of faces on P. As each face of P is processed only once, the overall time required by Step 2 is $O(\Sigma(k_i')\ (i = 1, 2, \ldots, m))$. As for the loop at Step 3, note that the indices of the faces of $\mathbf{\Omega}_i$ adjacent to edge e_j are stored in the order of their spatial angles in an entry of the E_f list of $\mathbf{\Omega}_i$. Therefore, only $O(\log k_j)$ time is needed to locate the position of f in that entry, and, hence, the index f' of the face well adjacent to face f at an edge e_j. As a result, the overall time taken by Step 3.1 is $O(\Sigma(k_i' \log k_i)\ (i = 1, 2, \ldots, m))$. The total time cost of MWCS_FACES is therefore $O(n + (\Sigma(k_i' \log k_i)\ (i = 1, 2, \ldots, m)))$.

Before the complete algorithm for carrying out the MWCS separation is presented, it is necessary to clarify that, given the indices of n faces that form the boundary of an MWCS of $\mathbf{\Omega}_i$, only $O(n)$ time is needed to construct the pseudopolyhedron representation $\langle V, E, F, \text{NORM}, E_f \rangle$ of that MWCS, say P_i. To see this, note that all the V, E, F, NORM and E_f lists of P_i are readily available in the $\langle V, E, F, \text{NORM}, E_f \rangle$ of $\mathbf{\Omega}_i$. The only work needed besides the retrieval is the reindexation of the vertices, edges and faces of P_i once they are retrieved from $\mathbf{\Omega}_i$. For example, if only vertices $\{v_3, v_4, v_7, v_9, v_{15}\}$ are on P_i, and there is an edge on P_i whose entry in the E list of $\mathbf{\Omega}_i$ is $\langle 9, 4 \rangle$, then this edge will become $\langle 4, 2 \rangle$ in the E list of P_i, because vertices v_9 and v_4 now sit at the fourth and second positions of the V list of P_i. Analogously, if edges $\{e_2, e_7, e_{10}, e_{13}\}$ are on P_i, and P_i has a face stored in the F list of $\mathbf{\Omega}_i$ as $\langle 10, 7, 13 \rangle$, then this face will become $\langle 3, 2, 4 \rangle$, owing to the reindexation of $\{e_2, e_7, e_{10}, e_{13}\}$. Clearly, this reindexation process can be done in $O(n)$ time through simple index mapping. Let MWCS_OUTPUT be such a process, that takes as input a pseudopolyhedron $\mathbf{\Omega}_i$ and a list L of indices of the faces of $\mathbf{\Omega}_i$, and outputs the pseudopolyhedron representation of an MWCS of $\mathbf{\Omega}_i$ whose faces are those of $\mathbf{\Omega}_i$ with indices in L. Using both procedures MWCS_FACES and MWCS_OUTPUT, the algorithm given next performs the MWCS separation.

MWCS_SEPARATION algorithm

```
Algorithm MWCS_SEPARATION (Ω_i)
/*      Compute the MWCSs of a pseudopolyhedron Ω_i and output them     */
begin
(1)     unmark all the faces in the F list of Ω_i
(2)     while (there is a face f in F which is not marked) do
(2.1)       L ← MWCS_FACES(f, Ω_i)
(2.2)       call MWCS_OUTPUT(L, Ω_i)
(2.3)       mark all the faces with indices in L
        end do {Step 2}
end MWCS_SEPARATION
```

Lemma 3: The MWCS separation of a pseudopolyhedron $\mathbf{\Omega}_i$ with n_f faces can be done in $O(n_f \log n_f)$ time and $O(n_f)$ space.

Proof. In the MWCS_SEPARATION algorithm, Step 1 takes $O(n_f)$ time. For the while loop at Step 2, as each face can only be in one MWCS, the overall time cost of Step 2.2 and Step 2.3 is clearly $O(n_f)$. The time taken by each execution of the procedure MWCS_FACES is in the form of $O(n + (\Sigma(k_i' \log k_i)\ (i = 1, 2, \ldots, m)))$, where n and m are the numbers of the faces and edges on that particular MWCS, and k_i and k_i' are the numbers of the faces of $\mathbf{\Omega}_i$ and that MWCS adjacent to an edge of the MWCS, respectively. For the same reason that a face of $\mathbf{\Omega}_i$ can only be in one of its MWCSs, the sum of $\Sigma k_i'$ ($i = 1, 2, \ldots, m$) over all the edges of $\mathbf{\Omega}_i$ is $O(\Sigma k_i$ ($i = 1, 2, \ldots, n_e$)), where n_e is the total number of edges of $\mathbf{\Omega}_i$. Therefore, after the termination of the MWCS_SEPARATION algorithm, the overall time taken by Step 2.1 is $O(n_f + (\Sigma k_i) \log n_f)$, i.e. $O(n_f \log n_f)$, as Σk_i ($i = 1, 2, \ldots, n_e$) is $O(n_f)$. QED

With Theorem 1 and Lemma 3, the following is in order.

Theorem 2: Whether a pseudopolyhedron $\mathbf{\Omega}_i$ is strongly nonconvergent or not can be detected in $O(n \log n)$ time, where n is the number of faces of $\mathbf{\Omega}_i$.

It is worth noting that, in the ASV process, the MWCS_SEPARATION algorithm not only detects the strong nonconvergence of a deficiency $\mathbf{\Omega}_i$, but also constructs the MWCSs of the deficiency $\mathbf{\Omega}_{i+1}$. The pseudopolyhedron representation of the MWCSs can then be used for the subsequent convex-hull and difference operations, along the corresponding branches after $\mathbf{\Omega}_{i+1}$.

FAST DETECTION FOR ASV NONCONVERGENCE

An ASV series is nonconvergent if it has a nonconvergent deficiency $\mathbf{\Omega}_n$. A way to detect the nonconvergence of an ASV series is to check the nonconvergence of every deficiency in the series. The time required by such a detection sets an upper bound.

Theorem 3: It needs at most $O(n^2 \log n)$ time to decide whether the ASV series of a pseudopolyhedron $\mathbf{\Omega}$ is

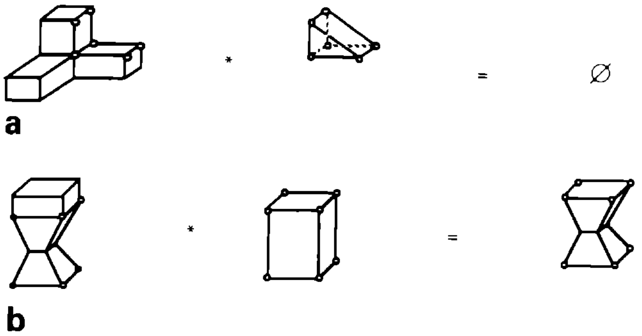

Figure 8. Regularized set intersection

convergent or not, where n is the number of vertices of $\mathbf{\Omega}$

Proof Recall that the difference operation is a vertex-elimination process. That is, a nonconvergent deficiency $\mathbf{\Omega}_k$ in ASV($\mathbf{\Omega}$) always has fewer vertices than $\mathbf{\Omega}_{k-1}$. In the worst case, suppose that only one vertex is eliminated after each difference operation To obtain the deficiency $\mathbf{\Omega}_k$ through the ASV process, k convex-hull and difference operations are needed, resulting in an overall time requirement of $\Sigma O(i \log i)$ $(i = n, n-1, \quad, n-k)$. Therefore, at most $\Sigma O(i \log i)$ $(i = n, n-1, \quad, 1) \leqslant O(n^2 \log n)$ time is needed to detect whether ASV($\mathbf{\Omega}$) converges or not. QED

In an attempt to improve this upper bound, the local cause of the ASV nonconvergence of a pseudopolyhedron $\mathbf{\Omega}$ is sought Such a study results in a sufficient condition for the ASV nonconvergence, which eventually leads to a linear detection algorithm In the search for this local cause, it is useful to invoke the mechanics of *regularized intersection*[5]

Definition 5 The regularized intersection of two pseudomanifolds **A** and **B**, denoted by $\mathbf{A} * \mathbf{B}$, is a pseudomanifold whose interior is $\mathbf{I}(\mathbf{A}) \cap \mathbf{I}(\mathbf{B})$

Figure 8 gives two examples of regularized intersection. As shown in Figure 8(a), the regularized intersection $\mathbf{A} * \mathbf{B}$ is the null set $\varnothing$, even though the ordinary set intersection $\mathbf{A} \cap \mathbf{B}$ yields two faces The result of the regularized intersection in Figure 8(b) is a nonconvergent pseudopolyhedron A tantalizing finding is revealed by the example in Figure 8(b) if there exists a (nonempty) subset (prior to the regularized intersection) that is nonconvergent, then the ASV series to be expanded is nonconvergent Such an observation is not a coincidence, the basis of this is shown by the following Lemma 4

Lemma 4. Let ζ be a subset of the vertices of a pseudopolyhedron $\mathbf{\Omega}$. If the regularized intersection between $\mathbf{\Omega}$ and CH(ζ) is a nonconvergent pseudopolyhedron, the ASV series of $\mathbf{\Omega}$ is nonconvergent.

Proof Assume that $\mathbf{\Omega} * \mathrm{CH}(\zeta)$ is a nonconvergent pseudopolyhedron It is claimed that all the vertices in ζ are noneliminatable Suppose that there is a deficiency $\mathbf{\Omega}_i$ in ASV($\mathbf{\Omega}$) whose vertex set is a superset of ζ, such that some vertex v in ζ is lost on the deficiency $\mathbf{\Omega}_{i+1}$ By Lemma 2, this means that all the incident faces of v are the hull faces of $\mathbf{\Omega}_i$ As CH(ζ) is a subset of CH($\mathbf{\Omega}_i$), it follows that v is also a weak hull vertex of $\mathbf{\Omega} * \mathrm{CH}(\zeta)$, which contradicts the assumption that $\mathbf{\Omega} * \mathrm{CH}(\zeta)$ is nonconvergent QED

Lemma 4 provides a sufficient condition for the nonconvergence of an ASV series, without the ASV process itself being invoked A direct implementation of such an algorithm is, however, infeasible, as there are $O(n!)$ subsets To reduce this high complexity, the characterization of local subsets of vertices, i e those that are adjacent to a common vertex, is investigated

Let two vertices of a pseudopolyhedron be said to be *adjacent* to each other if they are the two end points of an edge

Definition 6 A vertex v of a pseudopolyhedron $\mathbf{\Omega}$ is *supportable* if there exists a plane containing v such that the point set ξ_v lies on one side of that plane (where ξ_v consists of those vertices that are adjacent to v), otherwise, v is a *nonsupportable* vertex

As an example, all the vertices except for v of the pseudopolyhedron in Figure 9(a) are supportable. Also, as shown in Figure 9(b), a vertex v is nonsupportable if and only if it is strictly inside the convex hull of the vertices adjacent to it

Lemma 5 If a pseudopolyhedron $\mathbf{\Omega}$ has a nonsupportable vertex, then the ASV series of $\mathbf{\Omega}$ is nonconvergent

Proof Let v be a nonsupportable vertex of $\mathbf{\Omega}$, and ξ_v be the point set consisting of those vertices that are adjacent to v The lemma is proven by showing that $\mathbf{\Omega} * \mathrm{CH}(\xi_v)$ is a nonconvergent pseudopolyhedron As v is internal to CH(ξ_v), all its incident faces have portions that are internal to CH(ξ_v) Then, in each of these faces, there is a point that has an open 3D neighborhood that contains a subset of $\mathbf{I}(\mathbf{\Omega})$ that is strictly inside CH(ξ_v). By the definition of the regularized intersection, this neighborhood is preserved on $\mathbf{\Omega} * \mathrm{CH}(\xi_v)$ In other words, $\mathbf{\Omega} * \mathrm{CH}(\xi_v)$ must be a pseudopolyhedron, as its interior is not empty Now, consider a hull vertex p of $\mathbf{\Omega} * \mathrm{CH}(\xi_v)$, as illustrated in Figure 10(a) If p belongs to ξ_v, as none of the incident faces of v can be a hull face of CH(ξ_v), p can only be a strong hull vertex of $\mathbf{\Omega} * \mathrm{CH}(\xi_v)$ If p does not belong to ξ_v, it must be an intersection point between some face f of $\mathbf{\Omega}$ and a hull face of CH(ξ_v) (see Figures 10(b) and (c)) As face f has a portion internal to CH(ξ_v), which becomes an internal face of $\mathbf{\Omega} * \mathrm{CH}(\xi_v)$, p must be a strong hull vertex of $\mathbf{\Omega} * \mathrm{CH}(\xi_v)$. Therefore, all the

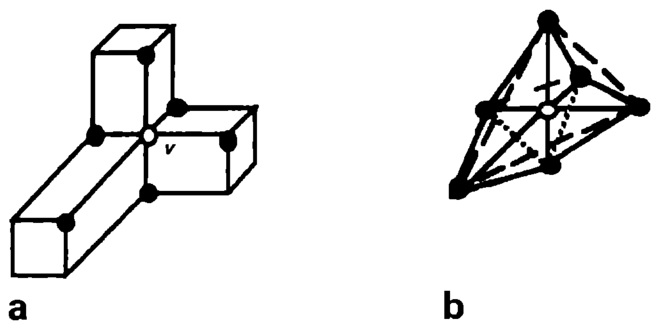

Figure 9 Supportable and nonsupportable vertices

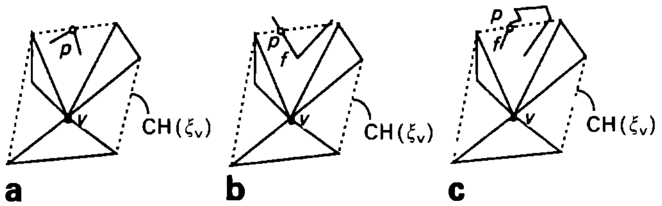

Figure 10 Proof of Lemma 6

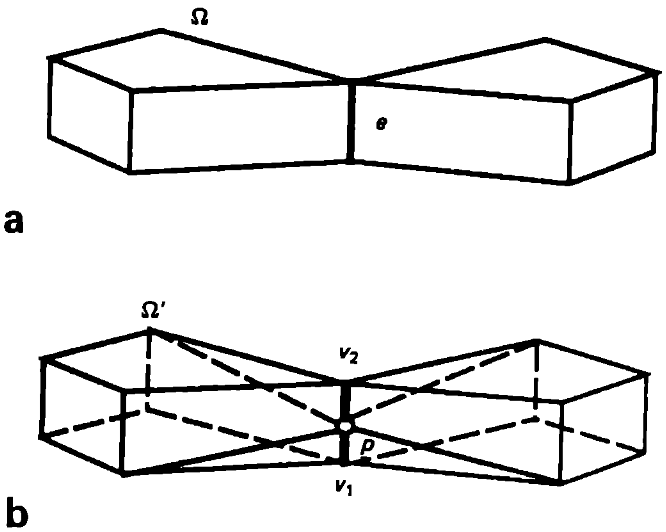

Figure 11 Nonsupportable vertex introduced by a nonsupportable edge

vertices of $\mathbf{\Omega} * CH(\xi_v)$ are either internal or strong. By Lemma 2, $\mathbf{\Omega} * CH(\xi_v)$ is nonconvergent. QED

As an illustration of Lemma 5, vertex v of the pseudopolyhedron in Figure 9(a) is nonsupportable. The regularized intersection between the pseudopolyhedron and $CH(\xi_v)$, where ξ_v are those six vertices adjacent to v, is another pseudopolyhedron, as shown in Figure 9(b), that is nonconvergent. By Lemma 5, the ASV series of the pseudopolyhedron in Figure 9(a) does not converge, this can easily be verified.

An extension to the supportability of vertices is the supportability of edges. Consider the pseudopolyhedron $\mathbf{\Omega}$ in Figure 11(a). Its ASV series can easily be shown to be nonconvergent, although all its vertices are supportable.

Definition 7. An edge e of a pseudopolyhedron is *supportable* if there exists a plane containing e such that all the faces incident to e are on one side of that plane, otherwise, edge e is nonsupportable.

Lemma 6. If pseudopolyhedron $\mathbf{\Omega}$ has a nonsupportable edge e, then ASV($\mathbf{\Omega}$) is nonconvergent.

Proof Assume that the nonsupportable edge e has k incident faces $f_1, f_2, \ldots, f_k$, and that v_1 and v_2 are its two vertices. Let p be an arbitrary point on e, but not v_1 or v_2. Also, let p_i, which is not v_1 or v_2, be a vertex on face f_i ($i = 1, 2, \ldots, k$). As the line segment $[p, p_i]$ is on a face f_i of $\mathbf{\Omega}$, the addition of p to the vertex set of $\mathbf{\Omega}$, and the addition of edges $[p, v_1]$, $[p, v_2]$, …, and $[p, p_i]$ ($i = 1, 2, \ldots, k$) to the edge set of $\mathbf{\Omega}$, introduces a new pseudopolyhedron representation of $\mathbf{\Omega}$, as shown in Figure 11(b). As e is nonsupportable, all the points in it, except, possibly, for v_1 or v_2, are strictly inside $CH(\{v_1, v_2, p_1, p_2, \ldots, p_k\})$. This implies that the vertex p is nonsupportable. By Lemma 5, ASV($\mathbf{\Omega}$) is nonconvergent. QED

It should be mentioned that Lemma 5 and Lemma 6 supply a sufficient but not necessary condition for nonconvergence. As an example, all the vertices and edges of the polyhedron in Figure 12 are supportable. However, its ASV series is nonconvergent.

Nevertheless, a linear time algorithm for detecting the sufficiency of nonconvergence offers an attractive alternative to the $O(n^2 \log n)$ time for both necessity and sufficiency.

Let o be the origin and (x_1, y_1, z_1), (x_2, y_2, z_2), …, (x_k, y_k, z_k) be k points in the 3D space. If the point o is supportable against (x_1, y_1, z_1), (x_2, y_2, z_2), …, (x_k, y_k, z_k), the angle between the normal vector $\mathbf{N}_o$ of a supporting plane P_o and the vector (x_i, y_i, z_i) must not be greater than 90° for all the $i = 1, 2, \ldots, k$ (see Figure 13). Conversely, if there exists a vector $\mathbf{N}_o$ such that the angle between it and a vector (x_i, y_i, z_i) ($i = 1, 2, \ldots, k$) is less than or equal to 90°, then the plane passing through o and orthogonal to $\mathbf{N}_o$ is clearly a supporting plane. Therefore, the detection of the supportability becomes the following: given k vectors (x_1, y_1, z_1), (x_2, y_2, z_2), …, (x_k, y_k, z_k), find another nonzero vector (A, B, C) such that $Ax_i + By_i + Cz_i \geqslant 0$ ($i = 1, 2, \ldots, k$). It is known[3] that the solution to this 3-variable problem, if it exists, can be obtained in $O(k)$ time.

Let SUPPORT(k, L) be a supportability detection procedure that takes a list L of k points as input, and outputs either 'true', if the origin is supportable against L, or 'false' otherwise. With the SUPPORT procedure, the following algorithm is in order.

NSV_DETECT algorithm

```
Algorithm NSV_DETECT(Ω)
/*  Detect the existence of nonsupportable vertices in a pseudopolyhedron Ω with
    n_v vertices
*/
begin
(1)     for i = 1, n_v do
(1.1)       v ← the ith vertex in the vertex list V of Ω
(1.2)       {p_1, p_2, ..., p_k} ← those vertices in V that are adjacent to vertex v
(1.3)       translate {p_1, p_2, ..., p_k} by a displacement of -v
(1.4)       if SUPPORT(k, {p_1, p_2, ..., p_k}) = 'false' then
                return with 'nonsupportable vertex found
            end if
        end do
(2)     return with 'no nonsupportable vertex found'
end NSV_DETECT
```

To devise an algorithm for detecting the supportability of an edge e of a pseudopolyhedron $\mathbf{\Omega}$, let v and v'

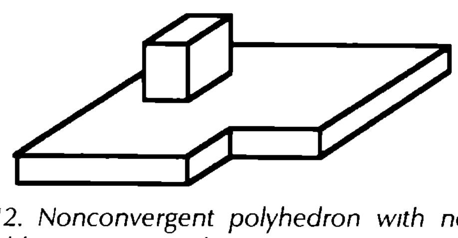

Figure 12. Nonconvergent polyhedron with no nonsupportable vertices or edges

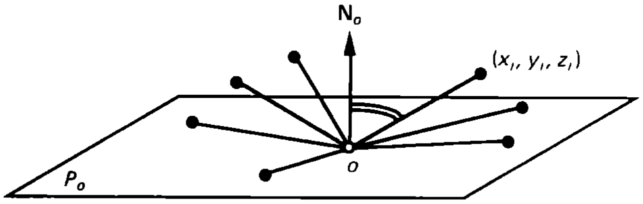

Figure 13. Angular relationship between the normal of a supporting plane and the adjacent vertices

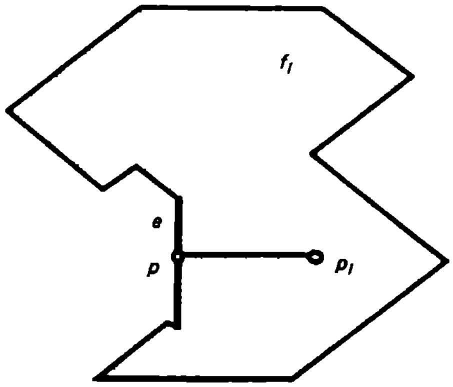

Figure 14 Finding the point p_i on face f_i

be the two end points of e, p be its center point, and $f_1, f_2, \ldots, f_k$ be the faces of $\mathbf{\Omega}$ incident to e Also, let p_i be a point on face f_i such that the line segment $[p, p_i]$ completely belongs to f_i ($i = 1, 2, \ldots, k$) (see Figure 14). Such a point p_i can be obtained in the constant time from the (clockwise or counterclockwise) order of the edges Let $FS(p, f_i)$ denote the function that returns the point p_i. Referring to the proof of Lemma 6, e is supportable if and only if p is supportable against the point set $\{v, v', p_1, p_2, \ldots, p_k\}$. This equivalence relationship gives rise to the following algorithm

NSE_DETECT algorithm

```
Algorithm NSE_DETECT(Ω)
 *   Detect the existence of nonsupportable edges in a pseudopolyhedron Ω with
     n_v vertices
 *
begin
(1)     for i = 1, n_e do
(1 1)       v, v' ← the two vertices of the ith edge in the edge list E of Ω
(1 2)       p ← the center point of [v, v']
(1 3)       f_1, f_2, ..., f_k ← those faces in Ω that are adjacent to the edge [v, v']
(1 4)       p_1, p_2, ..., p_k ← FS(p, f_1), FS(p, f_2), ..., FS(p, f_k)
(1 5)       translate {v, v', p_1, p_2, ..., p_k} by a displacement of −p
(1 6)       if SUPPORT(k + 2, {v, v', p_1, p_2, ..., p_k}) = 'false' then
                return with 'nonsupportable edge found
            end if
        end do
(2)     return with 'no nonsupportable edge found
end NSE_DETECT
```

Lemma 7 The existence of nonsupportable vertices and nonsupportable edges of a pseudopolyhedron $\mathbf{\Omega}$ with n_v vertices, n_e edges and n_f faces can be detected in at most $O(n_f)$ time

Proof The theorem is proven by showing that both the NSV_DETECT and NSE_DETECT algorithms run in $O(n_f)$ time For the NSV_DETECT algorithm, because the SUPPORT procedure runs in linear time, the time complexity required by the loop at Step 1 is linear in Σd_i ($i = 1, 2, \ldots, n_v$), where d_i is the degree of the ith vertex in $\mathbf{\Omega}$, which has been proven to be $O(n_f)$ in the Appendix of Part 1 of this paper Therefore, NSV_DETECT runs in $O(n_f)$ time For the NSE_DETECT algorithm, by similar reasoning, the time required by the loop at Step 1 is linear in Σk_i ($i = 1, 2, \ldots, n_e$), where k_i is the face adjacency index of the ith edge in $\mathbf{\Omega}$ In the Appendix of Part 1 of this paper, it has been shown that Σk_i ($i = 1, 2, \ldots, n_e$) is $O(n_f)$ Therefore, NSE_DETECT runs in $O(n_f)$ time QED

SUMMARY

It has been established that it takes $O(n^2 \log n)$ time to determine if the ASV series of a given $\mathbf{\Omega}$ converges In particular, it takes $O(n \log n)$ time to detect if a deficiency $\mathbf{\Omega}_i$ is nonconvergent To remedy the nonconvergence, an $O(n \log n)$ time algorithm is offered to separate the culprit deficiency $\mathbf{\Omega}_i$ into maximally well connected sets

As an expedient alternative to the $O(n^2 \log n)$ time detection for nonconvergence, the sufficiency condition for nonconvergence can be detected in $O(n)$ time.

REFERENCES

1 **Woo, T C** 'Feature extraction by volume decomposition' *Proc. Conf CAD/CAM Technology in Mechanical Engineering* Cambridge, MA, USA (1982)

2 **Tang, K and Woo, T** 'Algorithmic aspects of alternating sum of volumes Part 1 Data structure and difference operation' *Comput-Aided Des* Vol 23 No 5 (1991) pp 357–366

3 **Preparata, F P and Shamos, M L** *Computational Geometry* Springer-Verlag (1985)

4 **Chazelle, B M** 'Convex decomposition of polyhedra' *Proc. ACM Symp Theory of Computing* Milwaukee, USA (1981) pp 70–79

5 **Requicha, A A G and Voelcker, H B** 'Solid modeling current status and research directions' *IEEE Comput Graph & Applic* Vol 3 No 7 (1983) pp 25–37

BIBLIOGRAPHY

Baer, A, Eastman, C and Henrion, M 'Geometric modeling a survey' *Comput.-Aided Des* Vol 11 No 5 (1979) pp 253–272

Bollobas, C *Graph Theory* Springer-Verlag (1979)

Cary, A 'BUILD users' guide' *Document No 102* CAD Group, University of Cambridge, UK (Nov 1979)

Chiyokura, H and Kimura, F 'A method of representing the solid design process' *IEEE Comput Graph & Applic* Vol 5 No 4 (1985) pp 32–41

Preparata, F P and Hong, S J 'Convex hull of finite sets of points in two and three dimensions' *Commun ACM* Vol 20 No 2 (1977) pp 87–93

Requicha, A A G 'Representation for rigid solids: theory, methods and systems' *ACM Comput Surveys* Vol 12 No 4 (1980) pp 437–464

Requicha, A A G and Voelcker, H B 'Constructive solid geometry' *Production Automation Project TM-25* University of Rochester, USA (Nov 1977)

Weiler, K 'Topological structures for geometric modeling' *PhD Dissertation* Rensselaer Polytechnic Institute, USA (Aug 1986)

Woo, T C 'A combinatorial analysis of boundary data structure schemata' *IEEE Comput Graph. & Applic.* Vol 5 No 3 (Mar 1985) pp 19–27