

## 實體平面化之有限元素網格分割研究

計劃編號: NSC 89-2212-E-002-087

執行期限: 89年8月1日至90年7月31日

主持人: 尤春風 you@w3.me.ntu.edu.tw

執行單位: 臺灣大學機械系

+

### 摘要

本文以實體平面化資料為基礎, 提出一個自動的三維有限元素網格產生法則, 建立三維物體之有限元素。物體平面化的(SLA)資料廣用於快速成型機的加工。本文以實體平面化產生的資料, 搜尋網格節點與建立有限元素, 完成三維物體之網格分割。

在產生有限元素之過程中, 本法完全避開複雜的布林運算, 藉此提高有限元素產生之速度。所建立的有限元素, 可包含六面體元素與四面體元素, 以增進有限元素分析之準確度。本文並在各章附上數個測試實例, 以驗證法則之可行性。

**關鍵字:** 網格產生、實體平面化、六面體元素、四面體元素

In this paper an algorithm of automated mesh generation from STL format is presented. The algorithms are created from a set of triangular patches and then generated finite elements. Then three-dimensional mesh generation is accomplished. In order to reduce the computation time, the algorithm proposed here is not necessary to use the complex Boolean operation. The finite elements generated by this algorithm include the tetrahedrons and hexahedrons. Finally, some examples and a FEM analysis are illustrated to verify the feasibilities of the algorithm.

**Keywords:** mesh generation; triangular patch; tetrahedron; hexahedron

### 1. 前言

有限元素法 (FEM) 起源於40年代[1], 由於現今工業問題日益複雜, 以傳統解析法對物體進行分析, 已是不可能的事, 而另一方面, 由於電腦科技的日新月異, 以及在軟硬體上的突飛猛進, 使得數值方法作為解題途徑的有限元素法, 廣泛應用到各種領域上。

應用有限元素法分析前, 首先需將三維的物體進行網格分割, 亦即一般在有限元素法中, 所熟知的前處理。網格分割的好與壞, 直接影響有限元素法計算的結果, 故如何產生一個良好的網格分割, 將是一個重要的課題。

### 2. 文獻回顧

自動網格產生的法則, 截至今日已有多種方法被提出。

(1) 修正八元樹法: Yerry 和 Shephard[2]以空間分割的概念, 將物體用三個彼此互相正交的

平面, 分割為八個區域, 然後再測試此八個區域與原物體間的關係做進一步細分。

(2) 格點基礎的網格分割法: Schneiders[3]提出一個以格點為基礎的網格分割法, 此法則有兩個主要的步驟。首先是物體內部的網格分割, 其次為邊界網格的建立。

(3) 特徵基礎的網格分割法: Unruh and Anderson[4]提出以特徵為基礎的網格分割法。此法將物體分割為數個特徵的組合。而在有限元素產生的方面則是採用 Delaunay[5]三角演算法, 來建立四面體。

(4) 節點連接法: 此法與特徵基礎的網格分割法類似, 不同之處在於初始節點的產生上, 節點連接法是直接對物體的邊、面與體積產生節點, 而在有限元素的產生上, 亦是採用 Delaunay 三角演算法。

### 3. 研究內容

本文以物體平面化後的資料建立初始節點, 由這些初始節點產生物體之有限元素。物體平面化的(STL)資料廣用於快速成型機的加工。本文內容包括以下兩項: 網格節點之產生、有限元素之建立。

#### 3.1 索引座標系統

索引座標系統, 為節點位於此三維陣列中的位址, 亦即此節點在整個陣列中的索引值, 其三個座標軸以I、J、K表示。以空間中的圓柱體來說明, 參看圖1, B為空間中的圓柱體, 假設A是包覆著B的立方體包覆箱, 此立方體包覆箱的八個頂點, 左下角後方之頂點 $P_1$ , 其座標值在各個軸向上的分量均為最小值, 此頂點被定為索引座標系統之原點; 位於點 $P_1$ 對角位置上之頂點 $P_7$ , 此點座標值在各個方向上的分量, 均為包覆箱頂點中的極大值, 若各個軸向上所希望得到可能的最多元素個數, 分別為Isize、Jsize與Ksize, 則 $P_7$ 在索引座標系的座標即為(Isize, Jsize, Ksize)。

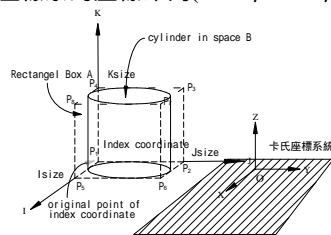


圖1 座標系統

### 3.2 節點產生

#### (1) 邊界節點組之搜尋

此步驟之目的在搜尋出物體所有的邊界節點，包含頂點節點、邊線節點、面節點與外部節點。或者說，邊界節點之搜尋條件，在於找到各方向上，距離邊界頂點、邊界線段與邊界面最接近之所有節點。

良好的邊界節點組，指的是一組能將索引座標系統裡的元素分成二個區域，各區域裡的元素均有屬於此區域裡的另一個元素與之相鄰，而二區域裡的元素彼此間則不相鄰。

#### (2) 內部節點之產生

內部節點搜尋方式為，以一良好邊界條件的任一節點為出發點，向前後左右上下六個方向搜尋，若搜尋到未指定屬性之節點，則將此節點指定為內部節點，並以此點做遞迴搜尋。

### 3.3 邊界節點之產生法則

#### 3.3.1 頂點節點之搜尋

頂點節點之搜尋是要搜尋所有的節點中，投影座標與三角形三個頂點之座標最接近之節點。

#### 3.3.2 邊線節點之搜尋

對一個三角形而言，當三個頂點節點確定後，便可以開始進行邊線節點之尋找，方法如下：

##### (1) 索引搜尋法

索引搜尋法是以三角形之三頂點節點之索引座標，任取兩個座標值，以線性內插之方法求得其內差節點之索引值。內差點之點數，需視該方向索引值之差而定。內差點之真實座標，亦以線性內差的方式，求得兩頂點座標之內差值，指定給予節點之真實座標。此法之優點在於其速度較快，但是搜尋所得到的節點有可能不是最接近邊界者，適合有限差分網格使用。

##### (2) 座標搜尋法

座標搜尋法與索引搜尋法程序相反。首先須先找到三角形的三個頂點節點，任取兩節點，以節點之真實座標，線性內差求得內差節點之座標值，反求得投影座標值與此座標值最接近之節點。座標搜尋法搜尋得的節點一定是最接近邊界者，但此法在速度上較索引搜尋法慢很多。

#### 3.3.3 面節點之搜尋

搜尋的基本概念類似掃描線之原理：在圖2中，掃描線在索引座標系統裡由下往上掃描。由於邊線節點的取法，是對各個方向分別取節點，參看圖3，假設以J方向作為邊界節點的搜尋方向時，則在J方向的每一層中，必剛好存在兩個邊線節點  $P_{t1}$  與  $P_{t2}$ ，以這兩個節點之真實座標為起訖座標值，依線性內差求得內差座標值  $v_1$ 。以此內差座標值求得其對應的索引座標值  $id_1$  與  $id_2$  所對應節點之投影座標值  $pv_1$ ，再將  $pv_1$  以此三角形法向量分量最大的方向，投影至此三角

形上，求得其座標值  $v_2$  與  $v_2$  所對應的索引座標值  $id_2$ ，節點  $id_2$  所對應的投影座標  $pv_2$ 。最後，進行由於邊界節點之產生法則，是搜尋各個軸向上，距離邊界面最近的節點，這個條件限制了節點在空間中之位置，圖4中以二維的例子，來說明節點在空間中之活動範圍，圖5中顯示一個節點的座標所能存在的範圍，只可能在一個以此節點的投影座標為中心點且各邊邊長為網格尺寸大小的矩形內（對三維物體而言則為一個立方體）。

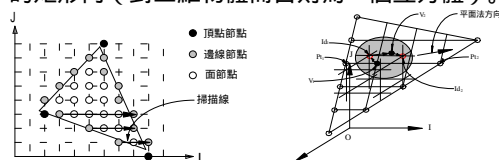


圖2 面節點搜尋之方法

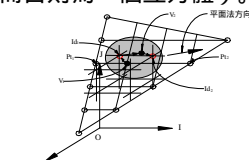


圖3 面節點搜尋法則

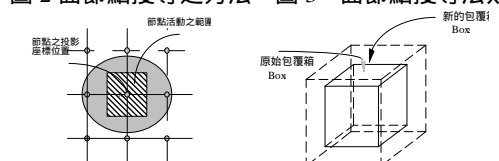


圖4 節點的活動範圍

圖5 新立方體包覆箱之建立

### 3.4 內部節點之產生法則

內部節點之產生必須先找到第一個內部節點。尋找的方法為將原來恰好包覆著物體的立方體包覆箱  $Box_1$ ，各面向外擴張一個網格單位大小的尺寸，形成一個新的立方體包覆箱  $Box_2$ ，如圖6所示，若將初始節點建立在  $Box_1$  裡，則位於最外層之初始節點必為外部節點，以任意一個外部節點為起始點，依搜尋內部節點的方式，對節點陣列進行搜尋，即可搜尋出所有的外部節點。再一次搜尋節點陣列，此時若搜尋到任意一個未指定屬性的節點，則此節點必為內部之節點，以此節點為第一個內部節點，進行內部節點之搜尋，即可找到所有位於物體內部之節點。

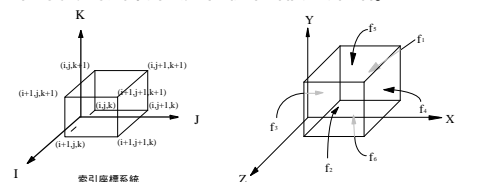


圖6 基礎六面體

圖7 基礎六面體與笛卡爾座標

### 3.5 有限元素之產生法則

基本網格單位，是在索引座標系統中，任意八個相鄰座標位置上之節點，所構成的一個資料群組。這八個座標位置所構成的六面體，稱之為基礎六面體(簡稱BH)如圖7所示。

有限元素的產生，首先需自節點陣列中搜尋出一個有效的基本網格單位。一個有效的基本網格單位需至少存在有四個節點資料且位於物體之內部。搜尋得到一個有效的基本網格單位，即

可開始由此基本網格單位建立有限元素，此部分包含從完整網格單位建立六面體元素與將破損網格單位分割成四面體元素二個部份，對於由破損網格單位所分割出來的四面體，需對其測試與原物體之內外關係，建立合理之四面體。本文中所產生的有限元素僅限於六面體與四面體，且為線性之元素，對於高階有限元素的建立，將不在本文的處理範圍之內。

### 3.5.1 有效的基本網格單位之搜尋

基本網格單位之搜尋是依序對索引座標系統中的三個座標軸進行。搜尋得到一個基本網格單位後，需檢查其節點的數目，對於節點數目小於四個節點的基本網格單位，由於此網格單位無法構成一個三維物體，所以必定為一個無效的基本網格單位。對於具有四個節點以上的基本網格單位，需要檢查此網格單位是否部份或全部位於物體之內部。檢測基本網格單位是否部份或全部位於物體之內部，可以下面四個方法完成：

- (1) 基本網格單位有一個以上的內部節點，則此基本網格單位必定有部份或全部位於物體之內部。
- (2) 基本網格單位的節點數目大於四且有一個以上的面節點，若與這些面節點在基本網格單位上共稜邊的所有節點，皆位於面節點所在的平面上或下方，則此基本網格單位亦存在部份或全部位於物體之內部。
- (3) 基本網格單位恰有四個節點，此四個節點不構成BH之任一平面且有一個以上的面節點，若非面節點之另三個節點至少有一位於面節點所在的平面上或下方，其餘則在面節點所在的平面上或下方，則此基本網格單位位於物體之內部。
- (4) 對於不屬於(1)、(2)與(3)之情形者，則視為無效的基本網格單位。

### 3.5.2 六面體元素之建立

一個有效的基本網格單位，若具有八個節點且為下列任一情況者，則此基本網格單位為一完整的網格單位。

- (1) 基本網格單位的所有節點均為內部節點，則此基本網格單位為完整的網格單位。
- (2) 基本網格單位六個面上的節點之真實座標，位在空間中的關係仍為平面，則此基本網格單位亦為一個完整的網格單位。

若搜尋得到的基本網格單位為完整的網格單位，則可由此網格單位建立六面體有限元素。

### 3.5.3 破損網格單位之切割

對於不滿足完整網格單位的基本網格單位，需將此基本網格單位切割成一組四面體元素的組合，所以此網格單位被稱為破損網格單位。

基本網格單位是建立是在基礎六面體上，對

一個基礎六面體，其在笛卡爾座標系統中亦為一個六面體，且此六面體各邊界面必存在笛卡爾座標系統的一個座標平面與之平行，此一座標平面稱為此六面體邊界面在笛卡爾座標系統中的投影面，如圖8中，基礎六面體之邊界面 $f_1$ 、 $f_2$ 、 $f_3$ 、 $f_4$ 、 $f_5$ 、 $f_6$ 中， $f_1$ 與 $f_2$ 之投影面為XY平面， $f_3$ 與 $f_4$ 之投影面為YZ平面， $f_5$ 與 $f_6$ 之投影面為ZX平面。若基礎六面體之任一平面上存在三個以上之網格節點，則此組節點將可構成基本網格單位之一個邊界面。定義此基本網格邊界面之投影面與基礎六面體之投影面相同。對此種基本網格邊界面定義其分割方法如下：

- (1) 若此邊界面恰存在三個節點，則由此三個節點組成破損網格的一個邊界面。
- (2) 若將此基本網格邊界面上之節點投影至其對應之投影面上，所構成的四邊形具有凹角，此時分割將以凹角出發，連接凹角所在的節點與不和其在基本網格單位上共稜邊的另一節點，將基本網格邊界面分割成二個三角形的網格邊界面。
- (3) 若基本網格邊界面的二個對角線，恰有一對角線其上的頂點不包含邊線節點外之其他屬性節點者，則以此對角線作為切割線段。
- (4) 若基本網格邊界面的二個對角線，恰有一對角線其上的頂點不包含頂點節點與邊線節點外之其他屬性的節點者，則以此對角線作為切割線段。
- (5) 對於不屬於上述之情形的破損網格邊界面則採取預設之分割方式。邊界面的預設分割方式如圖9。

若構成基本網格邊界面之節點，並非全部在基礎六面體之任一邊界面時此時分割方式如下：

- (1) 若此邊界面恰有三個節點，則由此三個節點組成破損網格的一個邊界面。
- (2) 若基本網格邊界面的二個對角線，恰有一對角線其上頂點不包含邊線節點外之其他屬性的節點者，則以此對角線作為切割線段。
- (3) 若基本網格邊界面的二個對角線，恰有一對角線其上的頂點不包含頂點節點與邊線節點外之其他屬性的節點者，則以此對角線作為切割線段。
- (4) 對不符合(1)、(2)與(3)之情形，任意連接二個不在基本網格單位共稜邊之頂點，將基本網格邊界面分割成二個三角形網格邊界面。

對破損網格單位各邊界面定義好分割方式，將分割完成之平面儲存於面陣列。面陣列即可在笛卡爾座標系統裡產生一個與破損網格單位對應的多面體。基於幾何理論，任何一個三維多面體，均可由有限個數的四面體組合成，此理論使得破損網格單位的分割處理得到保障。

Yerry and Shephard[2]將破損網格所構成之多面體上的各頂點，分成三邊共點、四邊共點與五邊共點三種，依序對各種頂點進行移除之程序，完成octant之切割。Shephard and George[6]提出三個運算元  $Op_1$ 、 $Op_2$  與  $Op_3$ 來將破損網格切割，如圖 9 所示：本文採用 Shephard and George 提出的元素移除運算元  $Op_1$ 、 $Op_2$  與  $Op_3$  作為建立四面體元素之法則。運用遞迴的方式，將破損網格中符合各元素移除運算元之四面體逐一移除，完成破損網格之切割。

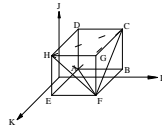
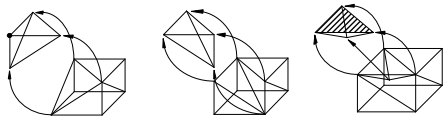


圖8 破損網格邊界面之預設分割方式



(a) vertex removal (b) edge removal (c) face removal  
圖9 元素移除運算元

### 3.5.4 外部四面體之移除

對破損網格單位所建立的四面體，不保證其必位於物體之內部。判斷四面體是否位於物體內部之法則如圖 10。此法則對四面體作進一步檢測，其結果能將必定位於物體外部四面體移除。

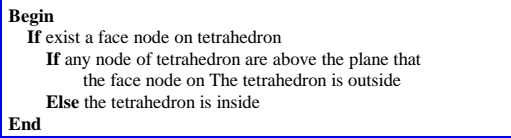


圖10有效四面體之判斷法則

### 3.6 實例測試

本節提出數個測試實例：

圖11為一立方體被二個圓柱體貫穿，輸入之SLA檔共包含228格平面，網格分割後所產生之節點個數與元素個數分別為1084個與360個，產生節點花費3秒，產生有限元素花費11秒。

圖12為一曲柄軸，輸入之SLA檔共包含332個平面，網格分割後所產生之節點個數與元素個數分別為1138個與1728個，產生節點花費6秒，產生有限元素花費20秒。

圖13為Schneiders[3]中之一測試實例，輸入之SLA檔共包含148個平面，網格分割後所產生之節點個數與元素個數分別為6362個與5722個，產生節點花費2秒，產生有限元素花費32秒。

圖14為澆注件模型，輸入之SLA檔共包含1888個平面，網格分割後所產生之節點個數與元素個數分別為18294個與34684個，產生節點花費189秒，產生有限元素花費278秒。

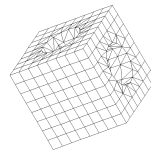


圖11六面體被二圓柱貫穿

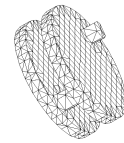


圖12曲柄軸網格分割



圖13Schneiders論文中之實例

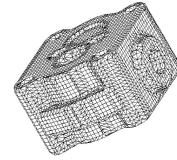


圖14澆注件模型

## 4. 結論

本研究對實體平面化資料，進行三維物體網格分割之工作，所得結論如下：

- (1) 本法則在產生節點的過程中，完全避開節點之內外測試；在建立有限元素之過程中，亦捨棄以布林運算之方式，作為建立破損網格與四面體切割之途徑，藉此改善網格分割之速度。
- (2) 本研究之節點搜尋法則，可提供各種需要產生初始節點之網格產生法則所使用，如格點基礎之網格分割法。

## 參考文獻

- [1] Hrennikoff, A., "Solution of problems in elasticity by the frame work method", *Journal of Applied Mechanics*, Vol. 8, No.4, pp.169-175, Dec. 1941
- [2] Yerry, M. A., "Trends in engineering software and hardware, Automatic mesh generation for three-dimensional solids", *Computers & Structures*, vol.20, No.1-3, pp.31-39, 1985
- [3] Schneiders, R., "A grid-based Algorithm for the generation of hexahedral element meshes", *Engineering with Computers*, vol.12, pp.168-177, 1996
- [4] Unruh, V., Anderson, D. C., "Feature-based modeling for automatic mesh generation", *Engineering with Computers*, vol.8, pp.1-12, 1992
- [5] Chongjiang Du, "An algorithm for automatic Delaunay triangulation of arbitrary planar domains", *Advanced in Engineering Software*, vol.27, pp.21-26, 1996
- [6] Shephard, M S., George, M. K., "Automatic three-dimensional mesh generation by the finite octree technique", *International Journal for Numerical Methods in Engineering*, vol.32, pp.709,741, 1991