

# 行政院國家科學委員會專題研究計畫 期中進度報告

## 適用於紙模型重建之網格參數化技術(1/2)

計畫類別：個別型計畫

計畫編號：NSC94-2213-E-009-087-

執行期間：94年08月01日至95年09月30日

執行單位：國立交通大學資訊工程學系(所)

計畫主持人：莊榮宏

報告類型：精簡報告

報告附件：出席國際會議研究心得報告及發表論文

處理方式：本計畫可公開查詢

中華民國 95 年 7 月 25 日

## 摘要

我們提出了一套針對紙模型重建需求之網格模型參數化機制。為了讓攤平後的網格模型可以重建三維立體模型，我們必須確保參數化過程中邊界長度之固定，並將整體參數化之最大誤差限制在一給定範圍內。為了達到上述需求，我們首先針對網格模型之邊界進行參數化，接著逐一將內部頂點由邊界向內進行攤平。當參數化過程中有任一頂點之映對誤差大於給上限，則對三維模型進行裁切並重新執行參數化動作。此一步驟將會重複執行直到所有頂點的映對誤差都在給定範圍內為止。由於我們在邊界參數化的過程中限制邊長為固定，因此邊界上的邊在重建的過程中可以避免產生無法對映的問題，且產生之參數化結果可以確保映對誤差小於一給定範圍。

### Abstract

We propose a mesh parameterization framework for paperfract design. The length of the boundary edges should be constrained during the parameterization process to ensure that the mesh after embedding can be reconstructed. Also, the maximum mapping error should be bounded after parameterization. To achieve these goals, we first apply the boundary parameterization method to parameterize the boundary vertices. Then the internal vertices are parameterized iteratively from the boundary. For any vertex with the mapping error exceeds the given upper bound, we cut the mesh by finding the shortest path between the vertex and mesh boundary and then reparameterize it. The entire process is proceed iteratively until no vertex has the mapping error exceeds the given threshold. While the length of boundary edges are constrained with their length, the result of parameterization can be reconstructed.

## 1. 前言

參數化的目的在於得到三維網格與二維平面間一對一的對應關係，以降低計算之複雜度。近年來，網格模型參數化的研究有不少成果被提出來，其應用相當廣泛，從網格重建(remeshing)、網格壓縮、形變(morphing)、模型編輯、與貼圖等，依據不同的應用，所需的參數化方式也有所不同。

紙模型為一種將三維模型攤平至二維平面之技術，並利用紙張可彎曲之特性將其重建。其中由三維模型攤平至二維平面即為參數化之動作，因此本計畫之目的在於找出合適的參數化方法來符合紙模型重建之需求。

Mitani 等人是第一個嘗試找出自動化紙模型設計之方法[1]。他們利用簡單的規則來對三維多邊形模型產生帶狀區域，並針對每條帶狀區域進行模型簡化動作。其方法雖然可以無失真的將攤平後的模型重建回簡化過的模型，卻不能控制模型簡化前後的幾何誤差。且 Mitani 等人的方法只能適用在高密度且平滑的三角形網格模型上，對於低密度或者是有明顯特徵邊緣的模型則無法達到好的效果。

雖然三角形網格模型可以透過裁切方式攤平在二維平面上，且不造成任何扭曲，但是此結果通常過於複雜以致於無法人工重建。而大部分的參數化方法是透過找最小誤差方式來求得對應關係，其誤差也可能過大以致於使得重建後的模型有很明顯的失真。且目前所知的參數化方法並沒有針對切割開的邊緣設限制，因此切割開的邊緣在模型重建後會產生明顯的破洞現象或完全無法粘合。Sorkine 等人提出一套可限制參數化最大誤差之方法[2]。透過從一種子三角形出發，逐一向外擴張至超過誤差上限為止，並由該區域開始將模型切割開來繼續擴張，直到所有三角形都攤平為止。雖然該方法可以限制住最大的誤差值，不過其最大誤差會落在網格參數化後的邊界上，使得參數化結果無法將被切割開的邊重建，因此無法適用於紙模型設計上。

我們提出了一套針對紙模型特點設計之網格模型參數化機制，使得參數化後之映對誤差可以被限制在一範圍內，且邊界與裁切

後的邊線長度可以被保留，使其可以被重建。我們所提出之系統分為以下三部分：網格邊界參數化、網格內部參數化、與網格裁切。首先透過網格邊界參數化來對邊界頂點之參數座標設限。之後再利用網格內部參數化來找出內部頂點之參數座標，並累積參數化過程之映對誤差值。當映對誤差大於給定上限，透過網格裁切機制對模型進行切割以降低誤差值。並重新執行邊界與內部參數化直到所有頂點之誤差值都小於給定上限。

## 2. 文獻探討

早期參數化的概念是將三維模型網格的幾何資訊，例如邊線的長度關係、邊與邊的角度關係、多邊形面積大小...等等，計算出到二維平面失真度最小的對映關係。

Maillot 等人首先提出以彈力系統理論 (elasticity theory) 為基礎，將多邊形的邊線視為彈簧，藉由計算其彈力網路最小能量，產生出二維對映之參數平面 [3]。Eck 等人提出利用 discrete harmonic map 之方式對於每個邊線給於不同權重係數，並將整個求解方式構建成一個線性系統，並透過解線性系統方式找到二維參數平面上的對應關係 [4]。Floater 利用 barycentric map 來建構線性系統，並給予合適的權重值來達到保留外觀 (shape-preserving) 的參數化效果 [5]。在 2003 年，Floater 又提出了另一種三角形座標的計算方式，稱為 mean-value coordinate [6] 以加快參數化求解之速度。

為了解決在 Floater 的方法中固定模型邊界所造成的失真，Hormann 與 Greiner 提出 MIPS 參數化方法，利用邊界的頂點可自由移動的方式，找出最佳的網格邊界參數座標 (natural boundary) 以降低其參數平面的失真程度 [7]。他們的做法是利用一誤差評量的方式來決定參數化前後每個三角形的誤差，並疊代式 (iterative) 的降低參數化所產生的總體誤差。但是缺點就是因為其最佳化的方法為非線性，也使得計算複雜度提升許多。Sheffer 等人利用衡量網格頂點一圈 (one-ring) 範圍內之角度變化方式，將參數化問題描述成一非線性系統，並透過牛頓法迭代方式進行計算 [8]。Levy 等人提出利用最小平方差近似的方法 (least-squares approximation)，將多邊形保

留角度特性的公式轉成一個線性系統，因此計算量較低 [9]。Desbrun 等人提出結合 discrete conformal map 與 discrete authalic map 的方式來做參數化 [10]。

Sander 等人針對三維模型網格提出了幾何扭曲評量 (geometry-stretch metric) 之方法 [11]。其方法延伸 MIPS 中所提出的誤差評量方式 [7]，計算出參數平面對映之三維模型網格的面積失真程度，並透過迭代 (iterative) 方式降低整體面積失真。

Sorkine 等人提出限制扭曲之網格參數化 (Bounded-distortion piecewise mesh parameterization) 來將參數化所造成的失真控制在一定範圍內 [2]。其作法由一種子三角形出發，逐一將相鄰的三角形加入參數平面中。由於三維網格中的一頂點對每個攤平後的三角形可能會有不同的位置，因此 Sorkine 等人利用區域性的鬆弛方式 (local relaxation) 將頂點調整至最適當的位置，其中調整用的權重值則利用 Sander 等人所提出的幾何扭曲評量方式 [11] 來測量。

在紙模型設計上，Mitani 與 Suzuni 首先將多邊形網格模型拆解成若干帶狀區域方式來攤平，並配合模型簡化方式來降低重建時的複雜度 [1]。Julius 等人利用 developability 的資訊來進行模型拆解，並可用於後續的重建工作 [12]。其主要概念為將 developability 的資訊構成一評量標準，並在模型拆解時利用該資訊來判斷模型上的兩部分是否適合組成一個區塊 (chart)。

## 3. 研究方法

我們提出了一套針對三角形網格模型進行紙模型重建之參數化方法。此一方法分為三部分，首先針對網格邊界進行參數化。接著將每個內部三角形由邊界向內逐一攤平在二維平面上，並累積其攤平之誤差。當累積誤差大於一上限值時，我們找出造成該誤差之三角形頂點至網格邊界之最短路徑並將模型裁切開來，並重新進行網格參數化。此一步驟將重複進行直到網格模型上的所有三角形都被攤平至二維平面，且其累積誤差小於上限值為止。圖 1 為我們所提出之架構的流程圖。以下將針對此三部分進行深入介紹。

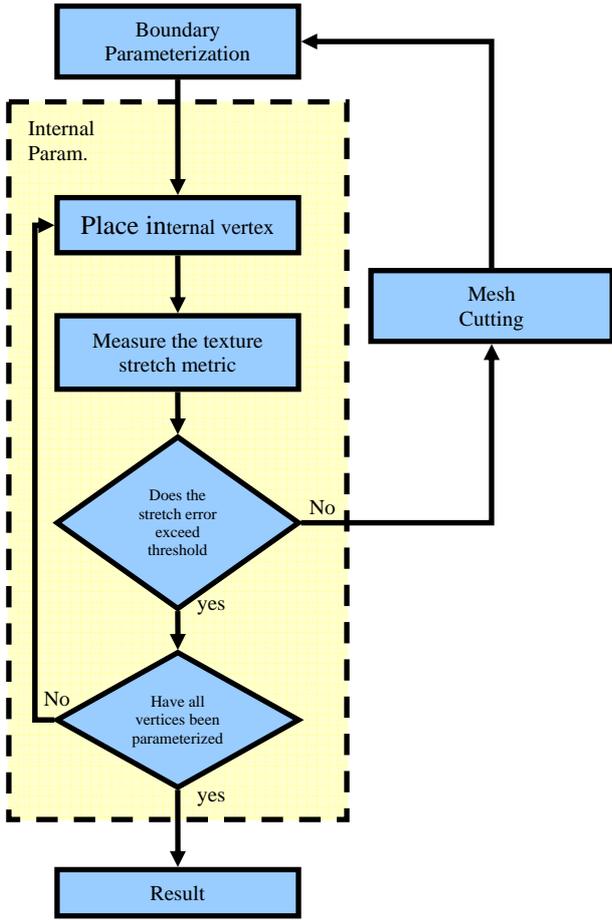


圖 1：系統流程圖。

### 3.1 網格邊界參數化

為了達到紙模型重建之目的，我們必須確保攤平後的網格邊界之長度與三維網格上的長度是一致的。目前已提出的網格參數化機制多半是將內部頂點之參數座標最佳化以降低攤平後的面積或角度誤差，此類方法並無法保證網格邊界之長度固定。因此我們提出了一套針對網格模型邊界之參數化機制。針對三維網格模型之邊界，我們設定其邊界長度為必須保持之限制，在參數化過程中設法降低三維模型邊界上的頂點角度與攤平後之二維平面上的角度差，透過解有限制之數值最佳化方法求出三維網格模型邊界上的頂點在二維平面上的最佳位置。

我們參考 Sheffer 等人所提出之網格模型參數化機制[8]，加入邊長之限制，並將其延伸至處理網格邊界之參數化。我們透過解有限制之數值最佳化方式找出邊界參數化，為了對三維模型與二維平面上的頂點角度差計

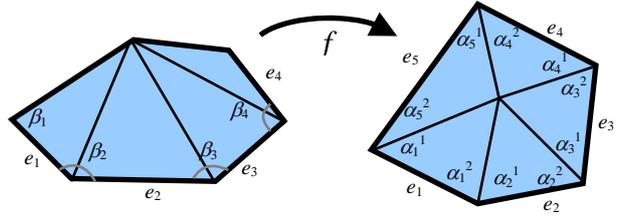


圖 2：邊界角度與映對示意圖。

算相鄰之最佳解，我們將三維網格簡化為一頂點與其一圈(one-ring)範圍之型式，即將三維網格中的所有內部頂點用一頂點取代，如圖 2 所示，並設定欲求之目標函式為公式 1 之型式：

$$F(\alpha) = \sum_{i=1}^n \sum_{j=1}^2 (\alpha_i^j - \beta_i^j)^2 \quad (1)$$

其中  $\beta_i^j$  為三維模型邊界上的角度，而  $\alpha_i^j$  為攤平至二維平面後的角度。為了讓邊界可以圍成一圈且內部三角形在攤平後不會退化為線或點，我們必須對上述角度設定限制。以下公式 2-4 為加入之限制。

$$g_{i,j}^{(1)} = \alpha_i^j \geq \varepsilon_1 > 0 \quad (2)$$

$$g_i^{(2)} = \alpha_i^1 + \alpha_i^2 < \pi - \varepsilon_2 \quad (3)$$

$$g^{(3)} = \sum_{i=1}^n (\pi - \alpha_i^1 - \alpha_i^2) - 2\pi = 0 \quad (4)$$

$$g^{(4)} = \frac{\prod_{i=1}^n \sin \alpha_i^2}{\prod_{i=1}^n \sin \alpha_i^1} - 1 = 0 \quad (5)$$

其中公式 2、3、4 為確保三角形在參數化後仍為三角形，而公式 5 是用來保證網格邊界在參數化之後可圍成一圈。為了達到邊界長度的固定，我們又加入了一限制式：

$$g_i^{(5)} = \frac{\sin(\alpha_{i+1}^1 + \alpha_{i+1}^2) \sin \alpha_i^1}{\sin(\alpha_i^1 + \alpha_i^2) \sin \alpha_{i+1}^2} - \frac{e_{i+1}}{e_i} = 0 \quad (5)$$

其中  $e_i$  為邊界上邊  $i$  的長度。最後我們的目標函式可以改寫成以下格式：

$$F^*(\alpha) = F(\alpha) + \lambda_3 g^{(3)} + \lambda_4 g^{(4)} + \sum_{i=1}^n \lambda_{5i} g_i^{(5)} \quad (6)$$

其中 $\lambda_3$ 、 $\lambda_4$ 、與 $\lambda_{5i}$  分別為一固定的常數來讓該等式成立。而根據公式 3、4、6 我們可以建出一非線性系統如公式 7。且透過求該非線性系統之方式我們即可得到此參數化的最佳解。

$$\frac{\partial F^*(\alpha)}{\partial \alpha_i^j} = \frac{\partial F(\alpha)}{\partial \alpha_i^j} + \lambda_3 \frac{\partial g^{(3)}}{\partial \alpha_i^j} + \lambda_4 \frac{\partial g^{(4)}}{\partial \alpha_i^j} + \sum_{i=1}^n \lambda_{5i} \frac{\partial g_i^{(5)}}{\partial \alpha_i^j} = 0 \quad (7)$$

$$g^{(3)} = 0$$

$$g^{(4)} = 0$$

$$g_i^{(5)} = 0$$

我們利用牛頓法來解公式 7 之非線性系統。首先我們計算公式 7 之一階與二階偏微分，並給定 $\alpha_i^j$ 之初始值為 $\beta_i^j$ ，透過迭代方式將 $\alpha_i^j$ 值收斂到可容忍之範圍內。

在求得 $\alpha_i^j$ 後，二維平面上的邊界參數值

可以由 $\alpha_i^j$ 與 $e_i$ 來計算。由一初始邊 $e_0$ 出發，將其起始頂點設定為原點且終點為 X 軸上長度為該邊長的位置，之後將邊的仰角加入 $\alpha_0^2 + \alpha_1^1$ 值並計算下一個邊之終點座標，逐一計算頂點座標直到所有頂點都算完為止。

### 3.2 網格內部參數化

當網格邊界上的頂點參數座標都決定之後，接下來必須要對其內部頂點進行參數化。由於網格邊界的三角形上有兩頂點之參數座標已經固定，因此我們的參數化順序從網格邊界出發，逐一針對每個三角形上剩下之一頂點進行參數化。為了衡量參數化造成的誤差，我們使用 Sander 等人所提出的誤差衡量機制[11]。假設 $T = \Delta q_1 q_2 q_3$ 為一在三維座標的三角形且 $T' = \Delta p_1 p_2 p_3$ 為其在二維平面上的參數座標值，則該三角形由二維平面至三維空間的映對的偏微分為：

$$S_s = \frac{\partial S}{\partial s} = \frac{q_1(t_2 - t_3) + q_2(t_3 - t_1) + q_3(t_1 - t_2)}{2\langle p_1 p_2 p_3 \rangle}$$

$$S_t = \frac{\partial S}{\partial t} = \frac{q_1(s_2 - s_3) + q_2(s_3 - s_1) + q_3(s_1 - s_2)}{2\langle p_1 p_2 p_3 \rangle}$$

因此，該映對的誤差值為

$$\gamma_{\max} = \sqrt{\frac{1}{2} \left( (a+c) + \sqrt{(a+c)^2 + 4b^2} \right)}$$

$$\gamma_{\min} = \sqrt{\frac{1}{2} \left( (a+c) - \sqrt{(a+c)^2 + 4b^2} \right)}$$

其中 $a = S_s \cdot S_s$ 、 $b = S_s \cdot S_t$ 、 $c = S_t \cdot S_t$ 。而 $\gamma_{\max}$ 與 $\gamma_{\min}$ 的值分別代表將一單位長度透過該映對方式轉換到三維空間中的最大與最小長度。因此我們定義一三角形之參數化誤差如公式 8 所示。

$$D(T, T') = \max \left\{ \gamma_{\max}, \frac{1}{\gamma_{\min}} \right\} \quad (8)$$

針對由網格邊界出發的每個三角形，由於我們已經知道該三角形中兩個頂點的參數座標，因此剩下的頂點座標可由該三角形於三維空間中的角度值求得。但是因為三維網格並不一定為平面結構，因此一內部頂點之參數座標可能會根據其相鄰的三角形而得到不同的值。為了解決此一問題我們先設定該頂點為由鄰近三角形所得到之平均座標，並計算鄰近三角形之映對誤差，在透過區域性的放鬆(relaxation)方式調整該頂點一圈(one-ring)區域內的三角形的總誤差值至最低，由此得到之參數座標值即為該頂點的座標。

### 3.3 網格裁切

在網格內部頂點參數化的過程中，我們會計算三角形經過參數化之映對誤差。當此誤差值過大時表示攤平後的三角形並不能有效的表現原始幾何，因此我們必須降低該區域的誤差值。將網格模型進行裁切是一種有效率的降低區域誤差的方式。對於產生過大映對誤差的內部頂點，我們找出該點至網格

模型邊界之最短路徑，並沿著此路徑將網格模型切割開，並重新進行邊界參數化。由於邊界參數化對映對後的角度進行最佳化，並同時保留邊長，因此可以大幅降低該頂點之映對誤差。

我們使用 Sheffer 所提出之最小延伸樹 (Minimum Spanning Tree) 之方法計算內部頂點至邊界的最短路徑[13]。首先將產生過大映對誤差之頂點與所有邊界頂點設為樹的端點，接著透過一修改過的 Dijkstra 最短路徑演算法求出一延伸樹，其中我們設定邊界上的兩頂點之間的距離為 0。

在找出最短路徑後，我們將最短路徑上的所有邊切割開來，並重新計算其邊界參數化與內部頂點參數化。此一步驟將會重複執行至沒有任何內部頂點產生過大的映對誤差為止。

#### 4. 實作與討論

我們使用 C/C++ 搭配 OpenGL 來實作此系統，而計算網格邊界參數化所需要之非線性系統則利用牛頓法迭代方式計算。透過牛頓法來解非線性系統之方法如公式 9 所示：

$$x^{k+1} = x^k - \left[ \frac{\partial f(x^k)}{\partial x} \right]^{-1} f(x^k) \quad (9)$$

其中  $\left[ \frac{\partial f(x^k)}{\partial x} \right]$  為該非線性系統對每一變數偏微分所組成之矩陣，且  $x^k$  為第  $k$  次迭代後的結果，因此公式 9 為一線性系統。我們利用 Matlab 來解牛頓法中每一次迭代所組成之線性系統。

由於邊界參數化之非線性系統中，欲求之目標函式已經包含對每一項限制的一階偏微分，因此透過牛頓法來解該系統必須計算所有限制的一階與二階偏微分，且矩陣

$\left[ \frac{\partial f(x^k)}{\partial x} \right]$  本身大小為  $(3n+2) * (3n+2)$ ，如公式 10 所示，其中  $n$  為邊界上的頂點個數，因此線性系統本身極為複雜。

$$\left[ \frac{\partial f(x^k)}{\partial x} \right] = \begin{bmatrix} \frac{\partial F_{\alpha_1^k}(\alpha)}{\partial \alpha_1^k} & \frac{\partial F_{\alpha_2^k}(\alpha)}{\partial \alpha_1^k} & \frac{\partial F_{\alpha_3^k}(\alpha)}{\partial \alpha_1^k} & \frac{\partial F_{\alpha_4^k}(\alpha)}{\partial \alpha_1^k} & \frac{\partial F_{\alpha_5^k}(\alpha)}{\partial \alpha_1^k} & \frac{\partial F_{\alpha_6^k}(\alpha)}{\partial \alpha_1^k} & \frac{\partial F_{\alpha_7^k}(\alpha)}{\partial \alpha_1^k} & \frac{\partial F_{\alpha_8^k}(\alpha)}{\partial \alpha_1^k} \\ \frac{\partial F_{\alpha_1^k}(\alpha)}{\partial \alpha_2^k} & \frac{\partial F_{\alpha_2^k}(\alpha)}{\partial \alpha_2^k} & \frac{\partial F_{\alpha_3^k}(\alpha)}{\partial \alpha_2^k} & \frac{\partial F_{\alpha_4^k}(\alpha)}{\partial \alpha_2^k} & \frac{\partial F_{\alpha_5^k}(\alpha)}{\partial \alpha_2^k} & \frac{\partial F_{\alpha_6^k}(\alpha)}{\partial \alpha_2^k} & \frac{\partial F_{\alpha_7^k}(\alpha)}{\partial \alpha_2^k} & \frac{\partial F_{\alpha_8^k}(\alpha)}{\partial \alpha_2^k} \\ \frac{\partial F_{\alpha_1^k}(\alpha)}{\partial \alpha_3^k} & \frac{\partial F_{\alpha_2^k}(\alpha)}{\partial \alpha_3^k} & \frac{\partial F_{\alpha_3^k}(\alpha)}{\partial \alpha_3^k} & \frac{\partial F_{\alpha_4^k}(\alpha)}{\partial \alpha_3^k} & \frac{\partial F_{\alpha_5^k}(\alpha)}{\partial \alpha_3^k} & \frac{\partial F_{\alpha_6^k}(\alpha)}{\partial \alpha_3^k} & \frac{\partial F_{\alpha_7^k}(\alpha)}{\partial \alpha_3^k} & \frac{\partial F_{\alpha_8^k}(\alpha)}{\partial \alpha_3^k} \\ \frac{\partial F_{\alpha_1^k}(\alpha)}{\partial \alpha_4^k} & \frac{\partial F_{\alpha_2^k}(\alpha)}{\partial \alpha_4^k} & \frac{\partial F_{\alpha_3^k}(\alpha)}{\partial \alpha_4^k} & \frac{\partial F_{\alpha_4^k}(\alpha)}{\partial \alpha_4^k} & \frac{\partial F_{\alpha_5^k}(\alpha)}{\partial \alpha_4^k} & \frac{\partial F_{\alpha_6^k}(\alpha)}{\partial \alpha_4^k} & \frac{\partial F_{\alpha_7^k}(\alpha)}{\partial \alpha_4^k} & \frac{\partial F_{\alpha_8^k}(\alpha)}{\partial \alpha_4^k} \\ \frac{\partial F_{\alpha_1^k}(\alpha)}{\partial \alpha_5^k} & \frac{\partial F_{\alpha_2^k}(\alpha)}{\partial \alpha_5^k} & \frac{\partial F_{\alpha_3^k}(\alpha)}{\partial \alpha_5^k} & \frac{\partial F_{\alpha_4^k}(\alpha)}{\partial \alpha_5^k} & \frac{\partial F_{\alpha_5^k}(\alpha)}{\partial \alpha_5^k} & \frac{\partial F_{\alpha_6^k}(\alpha)}{\partial \alpha_5^k} & \frac{\partial F_{\alpha_7^k}(\alpha)}{\partial \alpha_5^k} & \frac{\partial F_{\alpha_8^k}(\alpha)}{\partial \alpha_5^k} \\ \frac{\partial F_{\alpha_1^k}(\alpha)}{\partial \alpha_6^k} & \frac{\partial F_{\alpha_2^k}(\alpha)}{\partial \alpha_6^k} & \frac{\partial F_{\alpha_3^k}(\alpha)}{\partial \alpha_6^k} & \frac{\partial F_{\alpha_4^k}(\alpha)}{\partial \alpha_6^k} & \frac{\partial F_{\alpha_5^k}(\alpha)}{\partial \alpha_6^k} & \frac{\partial F_{\alpha_6^k}(\alpha)}{\partial \alpha_6^k} & \frac{\partial F_{\alpha_7^k}(\alpha)}{\partial \alpha_6^k} & \frac{\partial F_{\alpha_8^k}(\alpha)}{\partial \alpha_6^k} \\ \frac{\partial F_{\alpha_1^k}(\alpha)}{\partial \alpha_7^k} & \frac{\partial F_{\alpha_2^k}(\alpha)}{\partial \alpha_7^k} & \frac{\partial F_{\alpha_3^k}(\alpha)}{\partial \alpha_7^k} & \frac{\partial F_{\alpha_4^k}(\alpha)}{\partial \alpha_7^k} & \frac{\partial F_{\alpha_5^k}(\alpha)}{\partial \alpha_7^k} & \frac{\partial F_{\alpha_6^k}(\alpha)}{\partial \alpha_7^k} & \frac{\partial F_{\alpha_7^k}(\alpha)}{\partial \alpha_7^k} & \frac{\partial F_{\alpha_8^k}(\alpha)}{\partial \alpha_7^k} \\ \frac{\partial F_{\alpha_1^k}(\alpha)}{\partial \alpha_8^k} & \frac{\partial F_{\alpha_2^k}(\alpha)}{\partial \alpha_8^k} & \frac{\partial F_{\alpha_3^k}(\alpha)}{\partial \alpha_8^k} & \frac{\partial F_{\alpha_4^k}(\alpha)}{\partial \alpha_8^k} & \frac{\partial F_{\alpha_5^k}(\alpha)}{\partial \alpha_8^k} & \frac{\partial F_{\alpha_6^k}(\alpha)}{\partial \alpha_8^k} & \frac{\partial F_{\alpha_7^k}(\alpha)}{\partial \alpha_8^k} & \frac{\partial F_{\alpha_8^k}(\alpha)}{\partial \alpha_8^k} \end{bmatrix} \quad (10)$$

我們目前的實作在不考慮邊界長度限制

$\left( \left[ \frac{\partial f(x^k)}{\partial x} \right] \right)$  大小為  $(2n+2) * (2n+2)$  的情況下

已經可以得到一最佳化的邊界參數化，但由於長度限制(公式 5)之方程式本身相對於其他限制複雜許多，因此目前的實作系統在加入長度限制後無法收斂，我們仍在試圖解決此問題。

#### 5. 參考文獻

- [1] Mitani, J. and Suzuki, H. "Making Papercraft Toys from Meshes using Strip-based Approximate Unfolding" In Proceeding of SIGGRAPH, 2004.
- [2] Sorkine, O., Cohen-Or, D., Goldenthal, R. and Lischinski, D., "Bounded-distortion piecewise mesh parameterization", In Proceeding of IEEE conference on Visualization, 2002.
- [3] Maillot, J., Yahia, H., and Verroust, A. "Interactive Texture Mapping" In Proceeding of SIGGRAPH, 1993.
- [4] Eck, M., Deroose, T., Duchamp, T., Hoppe, H., Lounsbery, M., and Stuetzle, W. "Multiresolution, Analysis of Arbitrary Meshes." In Proceedings of SIGGRAPH, 1995.

- [5] Floater, M. S. "Parametrization and Smooth Approximation of Surface Triangulations". In *Computer Aided Geometric Design*, vol. 14, 1997.
- [6] Floater, M. S. "Mean value coordinates". In *Computer Aided Geometric Design*, Vol. 20, pp.19-27, 2003.
- [7] Hormann, K., and Greiner, G. "MIPS: An Efficient Global Parameterization Method" In *Curve and Surface Design: Saint-Malo*, 2000.
- [8] Sheffer, A. and Sturler, E. "Parameterization of Faced Surfaces for Meshing Using Angle-Based Flattening" *Engineering with Computers*, 17 (3), 326-337, 2001.
- [9] Lévy, B. and Maillot, J. "Least Squares Conformal Maps for Automatic Texture Atlas Generation" In *Proceedings of SIGGRAPH*, 2002.
- [10] Desbrun M., Meyer, M., and Alliez, P. "Intrinsic parameterizations of surface meshes". In *Proceeding of Eurographics*, 2002.
- [11] Sander, P. V., Snyder, J., Gortler, S. J., and Hoppe, H. "Texture Mapping Progressive Meshes" In *Proceeding of SIGGRAPH*, 2001.
- [12] Julius, D., Kraevoy, V. and Sheffer, A. "D-Charts: Quasi-Developable Mesh Segmentation" In *Proceedings of Eurographics*, 2005
- [13] Sheffer, A. "Spamming Tree Seams for Reducing Parameterization Distortion of Triangulated Surfaces" In *Proceedings of Shape Modeling International*, 2002.