アイソパラメトリック四辺形要素を用いた離散曲面制御
Discrete surface control using isoparametric quadrilateral elements

○藤田　慎之輔 *1
Shinnosuke Fujita *1

*1 北九州市立大学大学院 国際環境工学研究科 准教授
Assoc. Prof., Graduate School of Environmental Engineering, The University of Kitakyushu, Dr. Eng.

キーワード：離散曲面　アイソパラメトリック四辺形要素　面勾配
Keywords：Discrete surface　Isoparametric quadrilateral elements　Surface gradient

1. 概要

近年，コンピュテーショナルデザイン技術の一般化に伴い，古典幾何学にとらわれない自由な形態を有する建築構造物が設計されるようになってきた．そのような複雑な曲面を扱う場合，スプライン関数などのパラメトリック曲面を用いてジオメトリを連続的に扱うのが一般的である．パラメトリック曲面は比較的高い形状表現の自由度を有し，一般に節点数よりも少ない数の制御点座標値で曲面の滑らかさを保持したまま形状制御が可能であるため，形状最適化をはじめとした formfinding 技術と相性が良く，主に RC の自由曲面シェル構造物などの形状表現に利用されてきた他 [1)]，一般的な CAD ／ CG ソフトウェアに標準搭載されている．一方で，パラメトリック曲面はあくまで連続関数であるので，当該関数に乗らない形状は当然ながら表現することができない．また，ラチスシェルなどを想定した場合には，実際の構造物としてはグリッドが存在し，グリッドの 1 つ 1 つに仕上げが施工される離散曲面である．四角形グリッドを想定した場合，1 つ 1 つの面がねじれ面になってしまうと施工性が大きく低下するため，施工性を考えれば連続曲面としてパラメトリックに扱うことが必ずしも適切ではない場合がある．

本研究では，将来的に四角形グリッドで構成されるラチスシェルのジオメトリデザインに適用することを想定し，四角形要素で構成される離散曲面の形状をパラメトリック曲面を用いることなく直接的に扱いながら意匠の要求する形状や施工性に優れた形状を生成するための，面勾配の変化量を用いた形状制御汎関数を考案する．様々な解析モデルに対して，指定点を通る制約のもとで当該汎関数を目的関数として最小化することで，建築家のイメージする曲面の近傍で，滑らかでかつねじれが少なく仕上げの施工しやすい曲面を生成可能であることを示す．

2. 四角形要素の面勾配

各四角形要素の面勾配を定義するために 4 角形要素内の任意の座標 x, y, z を頂点の座標 $x_i, y_i, z_i (i = 1, \cdots, 4)$ を用いて次式で表す．

$$x = \sum_{i=1}^{4} N_i x_i,\ y = \sum_{i=1}^{4} N_i y_i,\ z = \sum_{i=1}^{4} N_i z_i \tag{1}$$

ここで，$N_i (i = 1, \cdots, 4)$ は形状関数で 2 つのパラメータ ξ, η を用いて次のように定義する ($-1 \leq \xi \leq 1,\ -1 \leq \eta \leq 1$).

$$\begin{aligned} N_1 &= \frac{1}{4}(1-\xi)(1-\eta),\ N_2 = \frac{1}{4}(1+\xi)(1-\eta) \\ N_3 &= \frac{1}{4}(1+\xi)(1+\eta),\ N_4 = \frac{1}{4}(1-\xi)(1+\eta) \end{aligned} \tag{2}$$

x, y のパラメータに関する偏微分は次式で記述できる．

$$\begin{aligned} \frac{\partial x}{\partial \xi} &= \sum_{i=1}^{4} \frac{\partial N_i}{\partial \xi} x_i \quad \frac{\partial x}{\partial \eta} = \sum_{i=1}^{4} \frac{\partial N_i}{\partial \eta} x_i \\ \frac{\partial y}{\partial \xi} &= \sum_{i=1}^{4} \frac{\partial N_i}{\partial \xi} y_i \quad \frac{\partial y}{\partial \eta} = \sum_{i=1}^{4} \frac{\partial N_i}{\partial \eta} y_i \end{aligned} \tag{3}$$

ここに，

$$\frac{\partial N_1}{\partial \xi} = -\frac{1}{4}(1-\eta), \quad \frac{\partial N_1}{\partial \eta} = -\frac{1}{4}(1-\xi) \tag{4}$$

$$\frac{\partial N_2}{\partial \xi} = \frac{1}{4}(1-\eta), \quad \frac{\partial N_2}{\partial \eta} = -\frac{1}{4}(1+\xi) \tag{5}$$

$$\frac{\partial N_3}{\partial \xi} = \frac{1}{4}(1+\eta), \quad \frac{\partial N_3}{\partial \eta} = \frac{1}{4}(1+\xi) \tag{6}$$

$$\frac{\partial N_4}{\partial \xi} = -\frac{1}{4}(1+\eta), \quad \frac{\partial N_4}{\partial \eta} = \frac{1}{4}(1-\xi) \tag{7}$$

日本建築学会情報システム技術委員会
第43回情報･システム･利用･技術シンポジウム論文集，148-151，2020年12月，オンライン
Proceedings of the 43rd Symposium on Computer Technology of Information，
Systems and Applications，AIJ，148-151，Dec.，2020，Online

また，形状関数の x, y に関する偏微分については，ヤコビアンを

$$\boldsymbol{J} = \begin{bmatrix} \dfrac{\partial x}{\partial \xi} & \dfrac{\partial y}{\partial \xi} \\ \dfrac{\partial x}{\partial \eta} & \dfrac{\partial y}{\partial \eta} \end{bmatrix} \tag{8}$$

とすれば

$$\begin{bmatrix} \dfrac{\partial N_i}{\partial \xi} \\ \dfrac{\partial N_i}{\partial \eta} \end{bmatrix} = \boldsymbol{J} \begin{bmatrix} \dfrac{\partial N_i}{\partial x} \\ \dfrac{\partial N_i}{\partial y} \end{bmatrix} \qquad i = 1, \cdots, 4 \tag{9}$$

の関係が成り立つことから

$$\begin{bmatrix} \dfrac{\partial N_i}{\partial x} \\ \dfrac{\partial N_i}{\partial y} \end{bmatrix} = \boldsymbol{J}^{-1} \begin{bmatrix} \dfrac{\partial N_i}{\partial \xi} \\ \dfrac{\partial N_i}{\partial \eta} \end{bmatrix} \qquad i = 1, \cdots, 4 \tag{10}$$

により陽に計算できる．

このとき，z 座標に関する面勾配 $\nabla z(\xi, \eta)$ は次式で表される．

$$\nabla z(\xi, \eta) = \begin{bmatrix} \dfrac{\partial z}{\partial x} \\ \dfrac{\partial z}{\partial y} \end{bmatrix} = \begin{bmatrix} \displaystyle\sum_{i=1}^{4} \dfrac{\partial N_i}{\partial x} z_i \\ \displaystyle\sum_{i=1}^{4} \dfrac{\partial N_i}{\partial y} z_i \end{bmatrix} \tag{11}$$

以降，定式化の簡略化のために，

$$\frac{\partial z}{\partial x}(\xi, \eta) = z_x(\xi, \eta), \quad \frac{\partial z}{\partial y}(\xi, \eta) = z_y(\xi, \eta) \tag{12}$$

とおく．また，e 番目の要素に関する量であることを左下添え字で表すものとする．例えば，要素 e の $\xi = 1.0$, $\eta = 0.0$ の点における x 方向勾配は ${}_e z_x(1.0, 0.0)$ となる．

3. 扱う最適化問題と数値例題

パラメトリック曲面を用いずに，指定された点を通るなるべく滑らかな曲面を求めることを考える．初期形状はスパン 10m を 6×6 に分割した正方形の格子メッシュにランダムな鉛直方向座標を付与したものとし,4 隅の点 ([0,0,0], [10,0,0], [0,10,0], [10,10,0]) と中央の点 ([5,5,4]) に指定座標を与える (図 1).

なお，ここでは四角形要素は図のように規則的に並べられているものと仮定する (図 2)．隣同士の面の勾配が急激に変化しないような汎関数を与えれば，滑かな曲面が生成可能と考える．x, y 方向のメッシュの分割数を N とし，各要素の中央 ($\xi = \eta = 0.0$) の勾配を x, y 方向それぞれで計算した上で，隣同士の勾配差の 2 乗和を f_1 とすると，次のようになる．

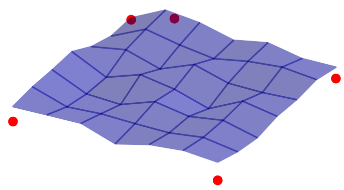


図 1　初期形状

6	12	18	24	30	36
5	11	17	23	29	35
4	10	16	22	28	34
3	9	15	21	27	33
2	8	14	20	26	32
1	7	13	19	25	31

図 2　要素番号図

$$\begin{aligned} f_1 = & \sum_{i=1}^{N}\sum_{j=1}^{N-1}\{{}_{N \cdot i+j}z_y(0,0) - {}_{N \cdot i+j+1}z_y(0,0)\}^2 \\ & + \sum_{j=1}^{N}\sum_{i=1}^{N-1}\{{}_{N \cdot i+j}z_x(0,0) - {}_{(N+1) \cdot i+j}z_x(0,0)\}^2 \end{aligned} \tag{13}$$

この式の第 1 項は y 方向について勾配差の 2 乗を計算しており，第 2 項は x 方向について勾配差の 2 乗を計算している．例えば，解析モデルの例だと，1-2 間，2-3 間，$\cdots$, 5-6 間，7-8 間，$\cdots$, 35-36 間の y 方向の勾配差の 2 乗和と，1-7 間, 7-13 間, $\cdots$, 25-31 間, 2-8 間，$\cdots$, 30-36 間の x 方向の勾配差の 2 乗和の和が f_1 となる．

まず, 汎関数 f_1 を最小化する次の最適化問題を解いてみる．

$$\begin{aligned} & \text{minimize} && f_1 \\ & \text{subject to} && z_i = \bar{z}_i \quad (i \in \mathrm{S}) \end{aligned} \tag{14}$$

ここで，S は z 座標を指定する節点番号の集合を表し，z_i は節点の z 座標，$\bar{z}_i$ は z 座標の指定値である．

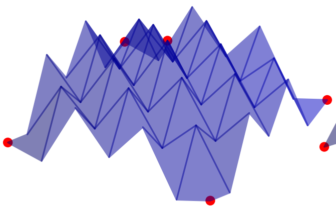
本論文で扱う最適化問題は全て逐次二次計画法 [2] により解くこととし，感度解析は差分近似により行った．図 1 の解析モデルに対して問題 (14) を解いた結果，最適性条件件を満たす解が得られたが，形状は図 3 のようになり，起伏が激しく滑らかさの失われた形状となった．四角形要素が極端なねじれ面となってしまっているため，それを抑制する汎関数として，各要素の対角の頂点における勾配差に着目し，その 2 乗和 f_2 を考える．

$$\begin{aligned} f_2 = \sum_{e=1}^{N^2} \Big[& \{{}_e z_x(-1,-1) - {}_e z_x(1,1)\}^2 \\ & + \{{}_e z_x(1,-1) - {}_e z_x(-1,1)\}^2 \\ & + \{{}_e z_y(-1,-1) - {}_e z_y(1,1)\}^2 \\ & + \{{}_e z_y(1,-1) - {}_e z_y(-1,1)\}^2 \Big] \end{aligned} \tag{15}$$

この汎関数を最小化すれば，ねじれ面が抑制される．先ほどの汎関数と合わせた次のような最小化問題を考える．

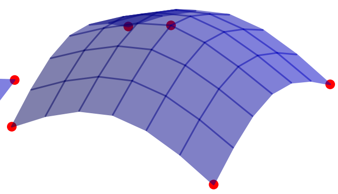
$$\begin{aligned} &\text{minimize} && f_1 + f_2 \\ &\text{subject to} && z_i = \bar{z}_i \quad (i \in \mathrm{S}) \end{aligned} \tag{16}$$

これを先ほどと同様に逐次二次計画法で解くと図 4 のように滑らかな曲面が得られた．



$f_1 = 2.644, f_2 = 941.6$

図 3 問題 (14) の最適化結果



$f_1 = 3.390, f_2 = 0.107$

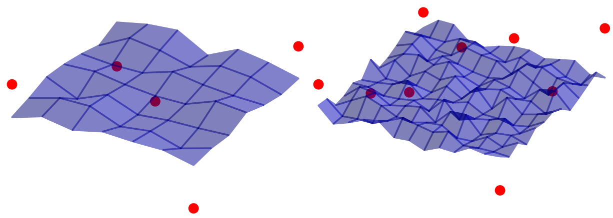
図 4 問題 (16) の最適化結果

4. その他の数値例題

問題 (16) の有効性と汎用性を確かめるため，指定点やグリッド分割数を変えた様々な数値例題を解く．設定した初期形状と指定点を図 5 に示す．モデル a, モデル b ともに初期形状はランダムに設定し，グリッド分割をそれぞれ 6x6, 12x12 としている．また，指定点はモデル a は 5 箇所，モデル b は 8 箇所設けた．

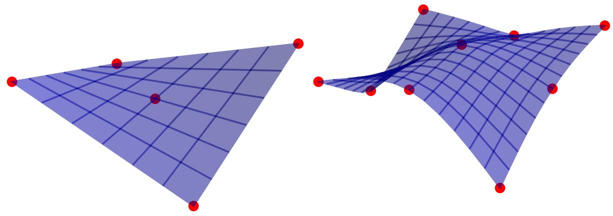
問題 (16) の最適化結果を図 6 にそれぞれ示す．また，図 7 及び 8 には，最適化結果の初期解や解析モデルの差異に対する依存性の有無を調べるために，初期解を様々に変化させて繰返し最適化計算を実行 (計 5 回) した際の収束履歴を f_1，f_2 のそれぞれの値について示している．

図 6 を見ると，モデル a, モデル b いずれの解析モデルについても，指定点をすべて通り，滑らかで，かつねじりの少ない離散曲面を生成できていることが確認できる．また，図 7 及び図 8 を見ると，モデル a, モデル b いずれの解析モデルも，初期解に依存することなく概ね同様の計算計算ステップ数で同一の最適解に収束していることがわかる．f_1 と f_2 は，それぞれマクロな滑らかさ，ミクロな滑らかさを制御する汎関数であると解釈できる．今回の数値実験では単純和を目的関数としたが，それぞれの汎関数に重み付けを行うことで，マクロな滑らかさ，ミクロな滑らかさのいずれを優先するかといった制御も可能であると考えられる．



(a) モデル a　　(b) モデル b

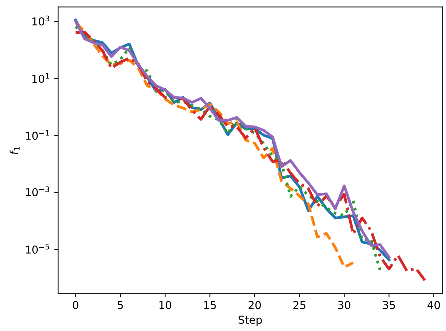
図 5　初期形状と指定点



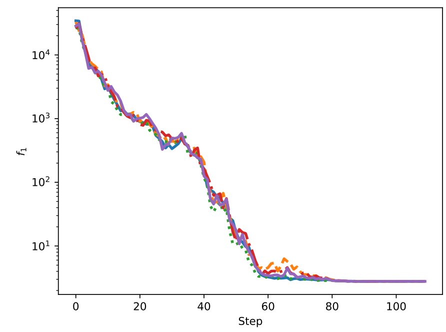
$f_1 = 0.0, f_2 = 2.560$　　$f_1 = 2.785, f_2 = 7.365$

(a) モデル a　　(b) モデル b

図 6　図 5 の解析モデルに対する問題 (16) の最適化結果

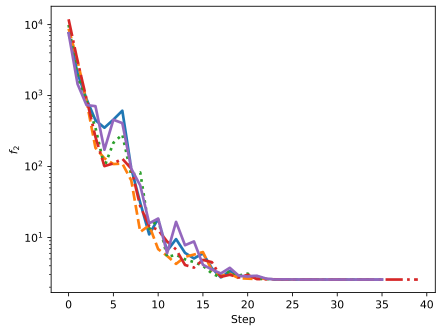


(a) モデル a

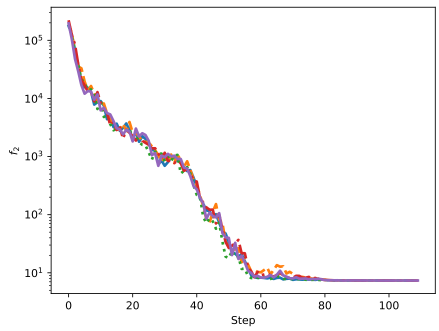


(b) モデル b

図 7　初期解を様々に再設定した場合の f_1 の収束履歴



(a) モデル a



(b) モデル b

図 8　初期解を様々に再設定した場合の f_2 の収束履歴

5.　まとめと今後の展望

パラメトリック曲面を用いずに，指定されて点を通り，かつ滑らかな曲面を生成するための形状制御汎関数を考案した．今回提案した汎関数はすべて連続関数であるので，数理計画法により高速に解くことが可能である．f_1 はマクロな滑らかさを，f_2 はミクロな滑らかさを測るものさしとして使用できる．なお，汎関数 f_2 は最小化すると曲面はマリオネットメッシュ[3)] に近づくと考えられる．前節で述べたように，それぞれの汎関数に重み付けを行うことで，マクロな滑らかさ，ミクロな滑らかさのいずれを優先するかを制御できる．また，仕上げ面の施工性を確実に担保するために，f_2 を制約条件として扱うアプローチも有効であると考えられる．

今後の展望としては，まずは定式化の汎用性向上が挙げられる．本研究で示した定式化は，x, y 方向に整然と並べられたトポロジーを前提としており，汎用性がなく数学的に美しくないため，あらゆるトポロジーに対して適用可能な仕組みの構築が望まれる．また，本手法を構造最適化問題へ組み込むことも容易であると考えられる．今回の数値実験では構造性能は一切考慮していないので，剛性 (ひずみエネルギー，コンプライアンス，etc.) や耐力 (応力分布，座屈荷重係数，etc.) といった力学的性能との多目的最適化への展開が期待される．

謝辞

本研究の一部は CREST JPMJCR1911 の助成を受けたものである．ここに記して謝意を表する．

[参考文献]

1）S. Adriaenssens, P. Block, D. Veenendaal, and C. Williams. Shell structures for architecture: Form finding and optimization. *Routledge*, 2014.

2）R. E. Perez, P. W. Jansen, and J. R. R. A. Martins. pyopt: a python-based object-oriented framework for nonlinear constrained optimization. *Structural and Multidisciplinary Optimization*, Vol. 45, No. 1, pp. 101–118, 2012.1.

3）R. Mesnil, C. Douthe, O. Baverel, and B. Leger. Marionette meshes: Modelling free-form architecture with planar facets. *International Journal of Space Structures*, Vol. 32, pp. 184–198, 2014.