

# GSMAC 有限要素法における質量の集中化とデータの再構築

## Mass lumping and data reconstruction in GSMAC FEM

橋本 学, 慶大院, 横浜市港北区日吉 3-14-1, E-mail: hasimoto@tana.mech.keio.ac.jp

棚橋隆彦, 慶大理工, 横浜市港北区日吉 3-14-1, E-mail: taka@mech.keio.ac.jp

Gaku HASHIMOTO, Keio University, 3-14-1, Hiyoshi, Kohoku-ku, Yokohama, 223-8522, Japan

Takahiko TANAHASHI, Keio University, 3-14-1, Hiyoshi, Kohoku-ku, Yokohama, 223-8522, Japan

Mass lumping is often carried out in order to avoid calculating the inverse of large total mass matrix after discretizing Navier-Stokes equation by FEM, but causes important problem when the size of satellite elements is not equal. In this study we indicate that the location where mass is lumped is different from node and think about the solution by using data reconstruction which consists of estimating center of gravity in dual element and correcting the data at center of gravity in dual element to that at node. We verify that data reconstruction is effective method by analyzing advection diffusion equation numerically and adopt the method into GSMAC FEM.

### 1. 緒言

Navier-Stokes 方程式を有限要素法で離散化すると, 大きな全体の質量行列が現れる。この逆行列を計算することを避けるために, 質量の集中化を行うことが多い。GSMAC 有限要素法においてもこの方法が導入されている<sup>(1)</sup>。しかし, この質量の集中化は節点の有する衛星要素の大きさが異なる時に数値計算上の誤差を生じる。質量が集中化される場所が節点ではないことがその原因である。

本研究ではデータの再構築を用いて, 双 1 次要素を用いた Galerkin 有限要素法における質量の集中化の問題点を解決する方法について考える。

### 2. 理論

データの再構築は, データの構築点(双対要素の重心)の評価と重心のデータから節点のデータへの補正から成る。

簡単のため, 1 次元移流拡散方程式の離散化を例にデータの再構築法について説明する(図 1 参照)。1 次元移流拡散方程式は, 保存形の場合

$$\frac{\partial f}{\partial t} + \frac{\partial(fu)}{\partial x} = n \frac{\partial^2 f}{\partial x^2} \quad (1)$$

非保存形の場合

$$\frac{\partial f}{\partial t} + u \frac{\partial f}{\partial x} = n \frac{\partial^2 f}{\partial x^2} \quad (2)$$

である。ここで,  $f$  は物理量の値,  $u$  は移流速度,  $n$  は拡散係数である。非定常項, 保存形の移流項, 非保存形の移流項, 拡散項の離散化式はそれぞれ

$$\left\langle \frac{\partial f}{\partial t} \right\rangle_i = \frac{\frac{l_1}{2} \left\{ \frac{(l_1/3)f_{i-1} + (2l_1/3)f_i}{l_1} \right\} + \frac{l_2}{2} \left\{ \frac{(2l_2/3)f_i + (l_2/3)f_{i+1}}{l_2} \right\}}{\frac{l_1 + l_2}{2}} \quad (3)$$

$$= \frac{\frac{l_1}{2} \frac{\partial \bar{f}_a}{\partial t} + \frac{l_2}{2} \frac{\partial \bar{f}_b}{\partial t}}{\frac{l_1 + l_2}{2}} = \frac{\frac{l_1}{3} \frac{\partial \bar{f}_a}{\partial t} + \frac{l_2}{3} \frac{\partial \bar{f}_b}{\partial t}}{\frac{l_1}{3} + \frac{l_2}{3}} \equiv \left( \frac{\partial f}{\partial t} \right)_{g^1} \quad (4)$$

$$\left\langle \frac{\partial(fu)}{\partial x} \right\rangle_i = \frac{(\bar{f}u)_B - (\bar{f}u)_A}{\frac{l_1 + l_2}{2}} = \frac{\frac{f_i u_i + f_{i+1} u_{i+1}}{2} - \frac{f_{i-1} u_{i-1} + f_i u_i}{2}}{\frac{l_1 + l_2}{2}} \equiv \left( \frac{\partial(fu)}{\partial x} \right)_{g^3}$$

$$\left\langle u \frac{\partial f}{\partial x} \right\rangle_i = \frac{\frac{l_1}{2} \left( u_A \frac{f_i - f_{i-1}}{l_1} \right) + \frac{l_2}{2} \left( u_B \frac{f_{i+1} - f_i}{l_2} \right)}{\frac{l_1 + l_2}{2}} = \frac{\frac{l_1}{2} \left( u \frac{\partial f}{\partial x} \right)_A + \frac{l_2}{2} \left( u \frac{\partial f}{\partial x} \right)_B}{\frac{l_1 + l_2}{2}} \equiv \left( u \frac{\partial f}{\partial x} \right)_{g^2} \quad (5)$$

$$\left\langle n \frac{\partial^2 f}{\partial x^2} \right\rangle_i = n \frac{\left( \frac{f_{i+1} - f_i}{l_2} \right) - \left( \frac{f_i - f_{i-1}}{l_1} \right)}{\frac{l_1 + l_2}{2}} = n \frac{\left( \frac{\partial f}{\partial x} \right)_B - \left( \frac{\partial f}{\partial x} \right)_A}{\frac{l_1 + l_2}{2}} \equiv \left( n \frac{\partial^2 f}{\partial x^2} \right)_{g^3} \quad (6)$$

となる。上式において,  $N_i(x)$  を要素内線形の形状関数とする

$$\langle f \rangle_i = \frac{\int_{x_{i-1}}^{x_{i+1}} N_i f dx}{\int_{x_{i-1}}^{x_{i+1}} N_i dx} \quad (7)$$

であり, 保存形の移流項, 拡散項を離散化するとき部分積分が用いられている。各項の双対要素の重心はそれぞれ

$$x_{g^1} = \langle x \rangle_i = x_i + \frac{1}{3}(l_2 - l_1) = \frac{x_{i-1} + x_i + x_{i+1}}{3} \quad (8)$$

$$x_{g^2} = x_i + \frac{1}{2}(l_2 - l_1) = \frac{x_{i-1} + x_{i+1}}{2} \quad (9)$$

$$x_{g^3} = x_i + \frac{1}{4}(l_2 - l_1) = \frac{x_{i-1} + 2x_i + x_{i+1}}{4} \quad (10)$$

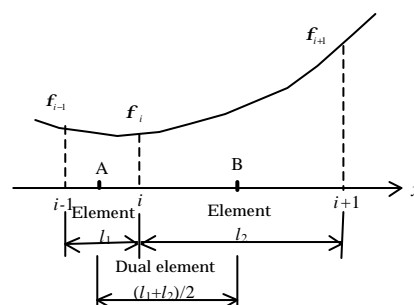


Fig. 1 Elements and dual element

である。非定常項は、要素内を線形近似して求めた  $f_a$  と  $f_b$  を加重平均することで構築された重心  $g^1$  のデータである。保存形の移流項は、要素内を線形近似して求めた  $(fu)_A$  と  $(fu)_B$  を双対要素内でさらに線形近似して構築された重心  $g^3$  のデータである。非保存形の移流項は、要素内を線形近似して求めた微分値と要素平均移流速度の積  $(u \partial f / \partial x)_A$  と  $(u \partial f / \partial x)_B$  を加重平均することで構築された重心  $g^2$  のデータである。拡散項は、要素内を線形近似して求めた微分値  $(\partial f / \partial x)_A$  と  $(\partial f / \partial x)_B$  を双対要素内でさらに線形近似して構築された重心  $g^3$  のデータである。そして、質量の集中化を行うと非定常項は

$$\left\langle \frac{\partial f}{\partial t} \right\rangle_i = \frac{\frac{l_1}{2} \frac{\partial f_i}{\partial t} + \frac{l_2}{2} \frac{\partial f_i}{\partial t}}{\frac{l_1}{2} + \frac{l_2}{2}} = \frac{\partial f_i}{\partial t} \quad (11)$$

となる。したがって、移流拡散方程式の離散化式は、保存形の場合

$$\left( \frac{\partial f}{\partial t} \right)_{g^1} = - \left( \frac{\partial (fu)}{\partial x} \right)_{g^3} + \left( n \frac{\partial^2 f}{\partial x^2} \right)_{g^3} \quad (12)$$

非保存形の場合

$$\left( \frac{\partial f}{\partial t} \right)_{g^1} = - \left( u \frac{\partial f}{\partial x} \right)_{g^2} + \left( n \frac{\partial^2 f}{\partial x^2} \right)_{g^3} \quad (13)$$

と評価される。さらに、質量の集中化を行うと、保存形の場合

$$\frac{\partial f_i}{\partial t} = - \left( \frac{\partial (fu)}{\partial x} \right)_{g^3} + \left( n \frac{\partial^2 f}{\partial x^2} \right)_{g^3} \quad (14)$$

非保存形の場合

$$\frac{\partial f_i}{\partial t} = - \left( u \frac{\partial f}{\partial x} \right)_{g^2} + \left( n \frac{\partial^2 f}{\partial x^2} \right)_{g^3} \quad (15)$$

と評価される。質量の集中化を行うことは、重心にデータが構築されている移流項、拡散項を使って節点と異なる点の値を時間進行していることを意味する。そこで、各重心での値から節点の値へ補正を行い、質量が集中する場所を節点にする。

### 3. 数値計算法

数値計算は以下のような手順で行う。

- (1) 各節点  $i$  に対応する重心  $g_i$  を計算する。
  - (2)  $t^n$  時刻の節点の値を使用し、重心における移流項、拡散項の値を求める。
  - (3) データの補正より、重心の値から節点の値を求める。
  - (4) 節点で時間進行し、 $t^{n+1}$  時刻の節点の値を求める。
- (2)から(4)の手順を繰り返すことで計算を進めていく(図2参照)。本研究では、データの補正には最小2乗法を用いる。3点  $g_{i-1}, g_i, g_{i+1}$  の値を用いて2次関数をつくり、節点  $i$  の値を求める。

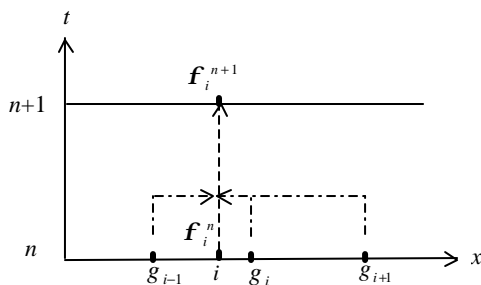


Fig. 2 Time marching by using data reconstruction

### 4. 結果・検討

#### (1) 拡散方程式

拡散方程式

$$\frac{\partial f}{\partial t} = n \frac{\partial^2 f}{\partial x^2} \quad (16)$$

を数値解析する。上式を

初期条件

$$t=0 \text{ で } f(x,0) = \sin px \quad (17)$$

境界条件

$$x=0 \text{ で } f(0,t) = 0 \quad (18)$$

$$x=1 \text{ で } f(1,t) = 0 \quad (19)$$

のもとで解くと、厳密解は

$$f(x,t) = \exp(-np^2 t) \sin px \quad (20)$$

で与えられる。

表1に計算条件を示す。使用するメッシュとしては、等分割したメッシュ1, 等比級数

$$x_i = \frac{L_x}{2} \frac{\exp\left(\frac{2a}{N}(i-1)\right) - 1}{\exp a - 1} \quad \left(i = 1 \sim \frac{N}{2}\right) \quad (21)$$

$$x_i = L_x - \frac{L_x}{2} \frac{\exp\left(\frac{2a}{N}(N+1-i)\right) - 1}{\exp a - 1} \quad \left(i = \frac{N}{2} + 1 \sim N + 1\right) \quad (22)$$

に従い分割したメッシュ2, 故意的に大小の要素が並ぶように分割したメッシュ3を用いる。メッシュ2において、 $L_x$ は領域の長さ、 $N$ は全要素数である。図3にCase1からCase8のメッシュ図を示す。ここで、大きさの異なる隣接要素の小さい方を大きい方で除した比を不等分割度と定義すると、その値はCase2で71.65%, Case3で51.34%, Case4で50.00%, Case6で84.65%, Case7で71.65%, Case8で50.00%である。Case1からCase8の全ての計算条件で  $a=1.0$  とする。結果として求まるのは対称な分布であるので、 $x$ が0.0から0.5の範囲について検討する。このとき、用いる相対誤差は

$$e = \frac{|f_{\text{approximate}} - f_{\text{exact}}|}{|f_{\text{exact}}|} \quad (23)$$

より算出する。

まず、理論で示したように  $\langle \partial^2 f / \partial x^2 \rangle_i$  の値を重心  $g^3$  の値と評価することの妥当性について考える。厳密解より  $f$  の  $x$  についての2階微分は

$$\frac{\partial^2 f}{\partial x^2} = -p^2 \exp(-np^2 t) \sin(px) \quad (24)$$

である。 $\langle \partial^2 f / \partial x^2 \rangle_i$  を重心  $g^3$  での値として評価した Case A,

$\langle \partial^2 f / \partial x^2 \rangle_i$  を節点  $i$  での値として評価した Case B,

$\langle \partial^2 f / \partial x^2 \rangle_i$  を重心  $g^3$  での値として評価し最小2乗法により

節点  $i$  での値を求めた Case Cの初期時刻  $t=0.0$ , 計算終了時刻  $t=0.1$  における相対誤差を図4, 図5に示す。図4, 図5から、不等分割度が大きいほど Case B と Case C の相対誤差は異なることが確かめることができる。そして、節点での Case B と Case C の値を比較することで、 $\langle \partial^2 f / \partial x^2 \rangle_i$  を重心  $g^3$  の

値として評価した方が節点  $i$  の値として評価するより、全体的に精度が良いことがわかる。ただし、図4(d)と(e)に見られるように領域中央付近において節点で評価した方が重心  $g^3$  で評価したより誤差が小さくなることもある。これは、ここでの衛星要素では要素内線形な関数で厳密解を十分に表すことができないことを意味する。以上より、 $\langle \partial^2 f / \partial x^2 \rangle_i$  の

値を重心  $g^3$  の値と評価することが妥当であると図4, 図5か

ら判断できる。

要素数 12, 24 の場合の厳密値, 質量の集中化を行う Galerkin 法による近似値, データの再構築を導入した Galerkin 法による近似値とそれらの相対誤差を図 6, 図 7 に示す。図 6 と図 7 からわかるように, 厳密解がなめらかな関数であるためにメッシュ 1 の結果よりメッシュ 2 で解析した方が相対誤差は大きくなる。これは厳密解がなめらかな関数であるために等分割で十分に関数の変化を捉えることができることや  $x=0.0$  へメッシュを寄せることで領域中央付近での衛星要素の大きさが大きくなり線形な関数でこの場所の厳密解を十分に表せないことが理由である。図 6 と図 7 において, データを再構築すると質量の集中化した場合と比べて領域内で相対誤差が小さくなる所と相対誤差が大きくなる所があることがわかる。このとき, 分布が線形とみなせるぐらいの大きさである不等分割度の大きい衛星要素に対してデータの再構築は有効であると考えられる。データの再構築により誤差が大きくなるのは, 図 4(b)と(c)に見られたような  $\langle \partial^2 f / \partial x^2 \rangle_i$  を重心の値とみなすことができない所の誤差の影響や陽的に時間進行している誤差の影響が理由であると考えることができる。

(2) Burgers 方程式による検証

Navier-Stokes 方程式のモデル方程式である Burgers 方程式

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2} \quad (25)$$

を数値解析する。上式を初期条件

$$t=0 \text{ で } u(x,0) = \sin px \quad (26)$$

境界条件

$$x=0 \text{ で } u(0,t) = 0 \quad (27)$$

$$x=1 \text{ で } u(1,t) = 0 \quad (28)$$

のもとで解くと, 厳密解は

$$u(x,t) = 4\nu n \frac{\sum_{n=1}^{\infty} n I_n \left( \frac{1}{2\nu n} \right) \sin(npx) \exp(-n^2 \nu t)}{I_0 \left( \frac{1}{2\nu n} \right) + 2 \sum_{n=1}^{\infty} I_n \left( \frac{1}{2\nu n} \right) \cos(npx) \exp(-n^2 \nu t)} \quad (29)$$

で与えられる<sup>(2)</sup>。ここで  $I_n$  は第 1 種変形 Bessel 関数である。式(29)において,  $t$  の増大と共に指数関数は 0 に近づく。そのため,  $u(x,0) = \sin px$  を十分に表している級数の項数をとり厳密値とみなす。本研究では, 全領域内で最大絶対誤差が  $4.921 \times 10^{-8}$  である 100 項までの級数をとる。

表 2 に計算条件を示す。使用するメッシュとしては, 図 3 と同様である。Case 1 から Case 8 の全ての計算条件で  $\nu=0.1$  とする。相対誤差の定義も式(23)と同様である。

要素数 12, 24 の場合の厳密値, 質量の集中化を行う Galerkin 法による近似値, データの再構築を導入した Galerkin 法による近似値とそれらの相対誤差を図 8, 図 9 に示す。図 8 と図 9 からわかるように, この場合も厳密解がなめらかな関数であるためにメッシュ 1 の結果よりメッシュ 2 で解析した方が相対誤差は大きくなる。

図 8 と図 9 においても, データを再構築すると質量集中化した場合と比べて領域内で相対誤差が小さくなる所と誤差が大きくなる所があることがわかる。図 8(a)と(d), 図 9(e)と(h)から, データの再構築を行うことで, 不等分割の相対誤差の分布が等分割の相対誤差の分布に近づいていることがわかる。

Tab. 1 Calculation condition

	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8
Mesh	Mesh 1	Mesh 2	Mesh 2	Mesh 3	Mesh 1	Mesh 2	Mesh 2	Mesh 2
Element number	12	12	12	12	24	24	24	24
		2.0	4.0			2.0	4.0	
$x_{min}$	$8.333 \times 10^{-2}$	$3.096 \times 10^{-2}$	$8.841 \times 10^{-3}$	$6.250 \times 10^{-2}$	$4.167 \times 10^{-2}$	$1.419 \times 10^{-2}$	$3.691 \times 10^{-3}$	$3.125 \times 10^{-2}$
$t$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-5}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-5}$	$1.0 \times 10^{-6}$	$1.0 \times 10^{-4}$
Time step number	$1.0 \times 10^3$	$1.0 \times 10^3$	$1.0 \times 10^4$	$1.0 \times 10^3$	$1.0 \times 10^3$	$1.0 \times 10^4$	$1.0 \times 10^5$	$1.0 \times 10^3$

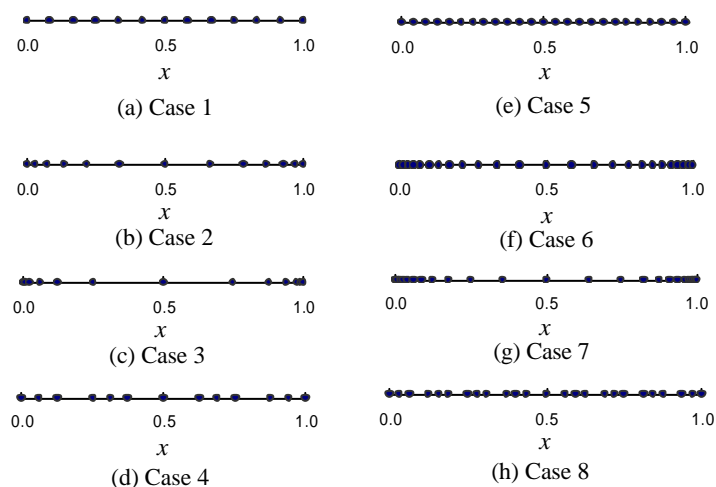
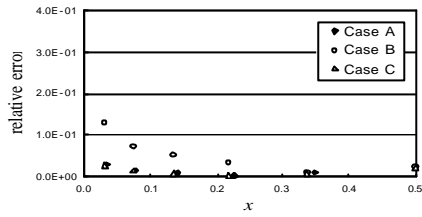
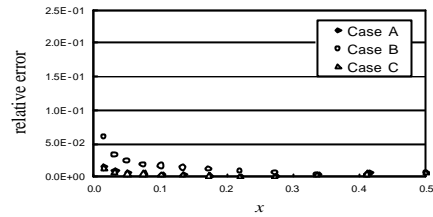


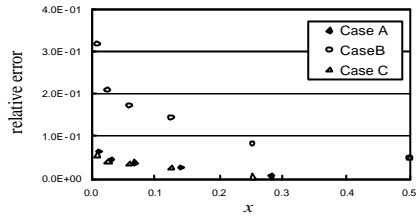
Fig. 3 Analysis mesh



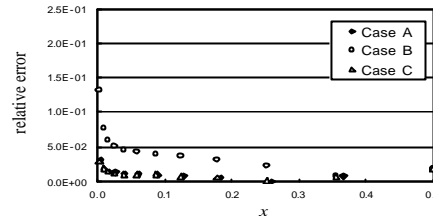
(a) Case 2



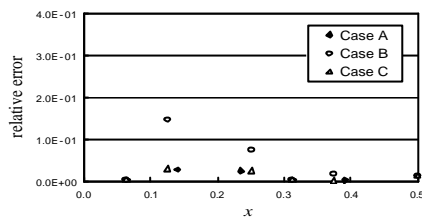
(d) Case 6



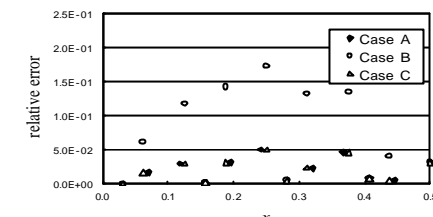
(b) Case 3



(e) Case 7

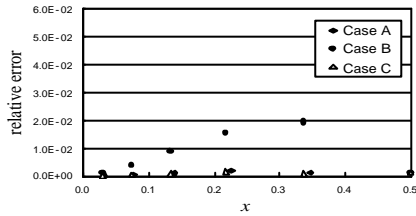


(c) Case 4

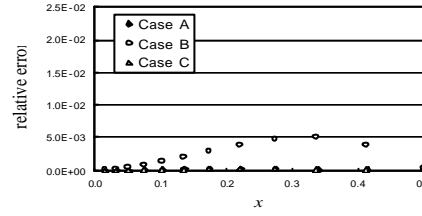


(f) Case 8

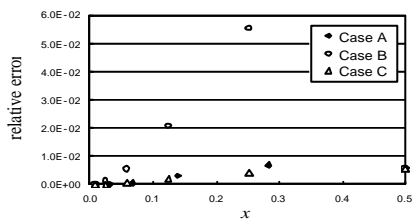
Fig. 4 The evaluation of  $\partial^2 f / \partial x^2$  at  $t=0.0$



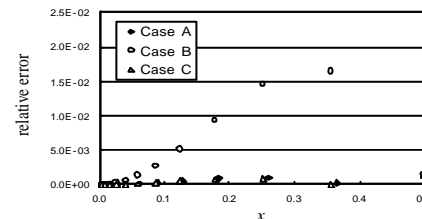
(a) Case 2



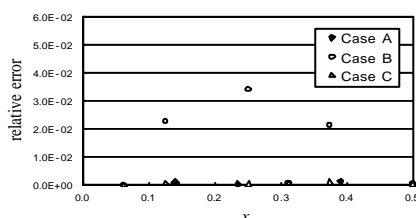
(d) Case 6



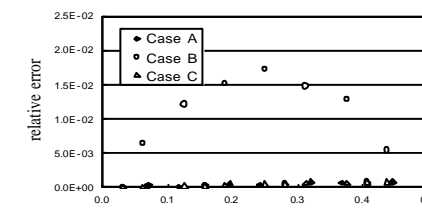
(b) Case 3



(e) Case 7

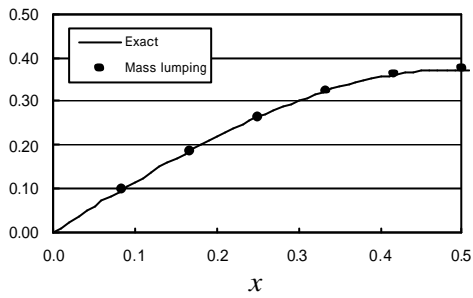


(c) Case 4

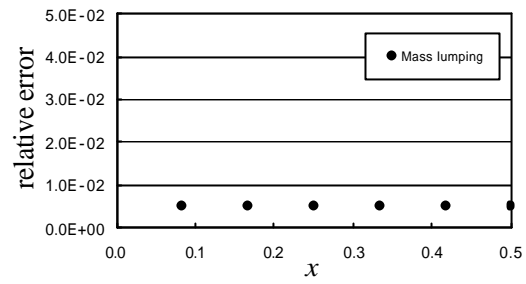


(f) Case 8

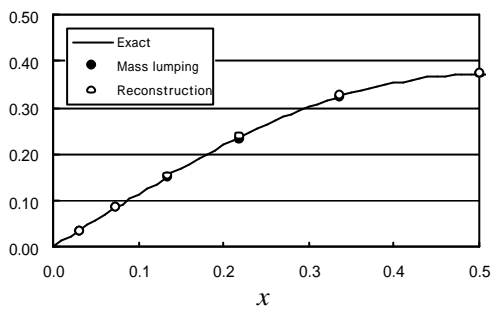
Fig. 5 The evaluation of  $\partial^2 f / \partial x^2$  at  $t=0.1$



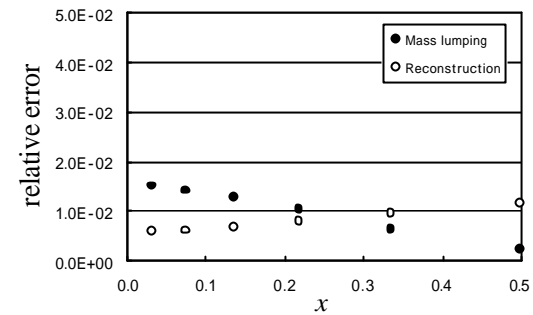
(a) Case 1



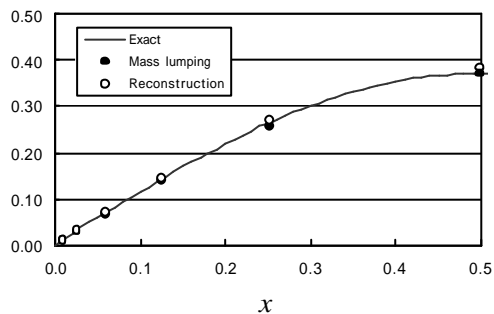
(a) Case 1



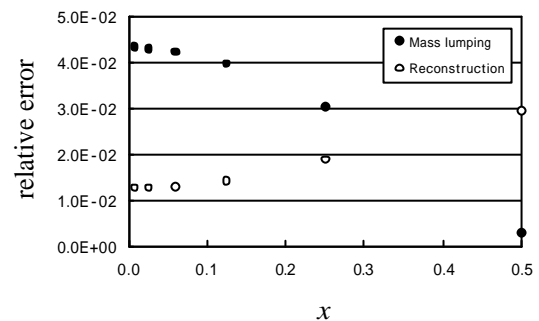
(b) Case 2



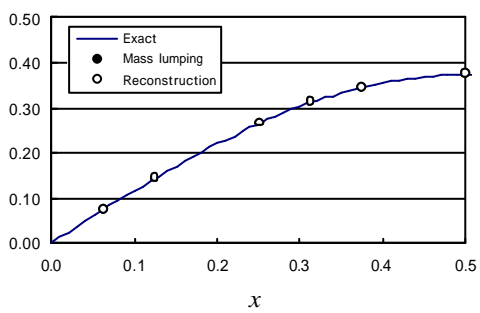
(b) Case 2



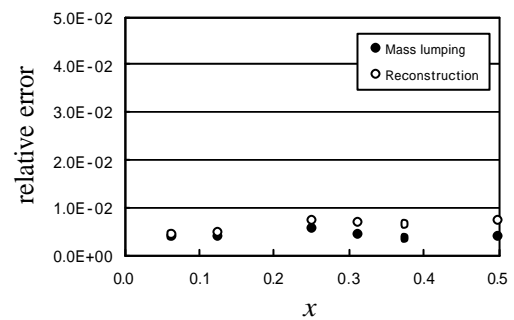
(c) Case 3



(c) Case 3

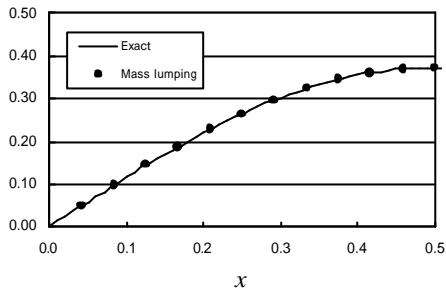


(d) Case 4

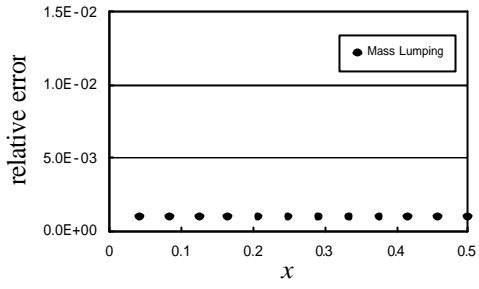


(d) Case 4

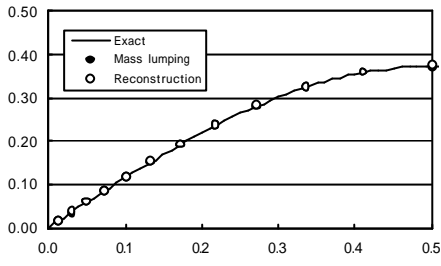
Fig. 6 The comparison between mass lumping and data reconstruction



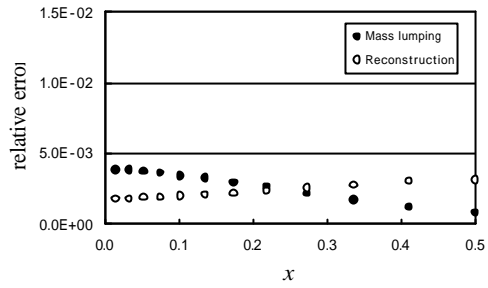
(e) Case 5



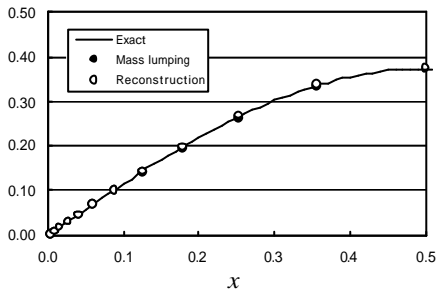
(e) Case 5



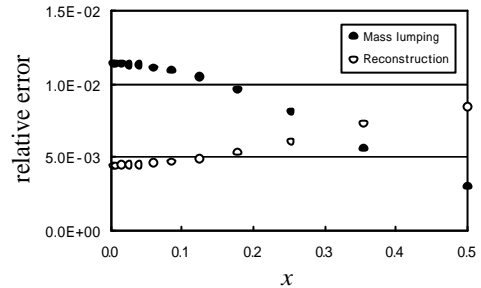
(f) Case 6



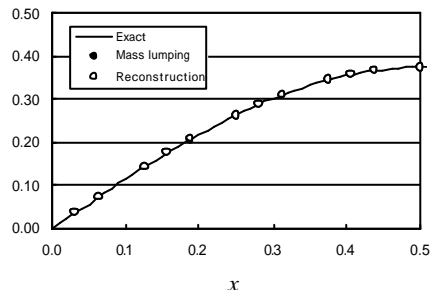
(f) Case 6



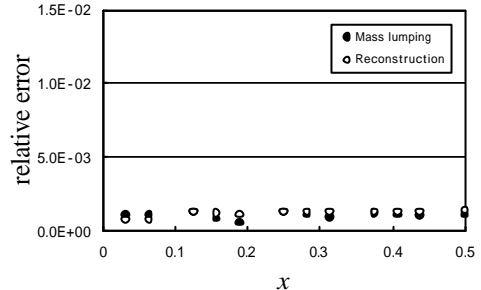
(g) Case 7



(g) Case 7



(h) Case 8

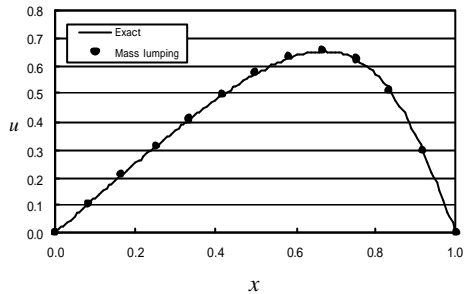


(h) Case 8

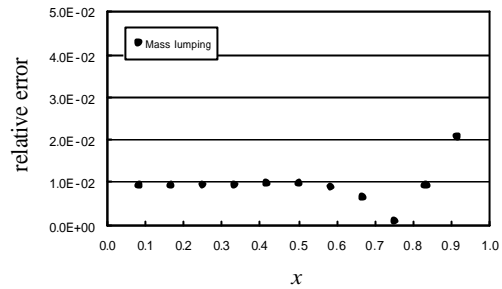
Fig. 7 The comparison between mass lumping and data reconstruction

Tab. 2 Calculation condition

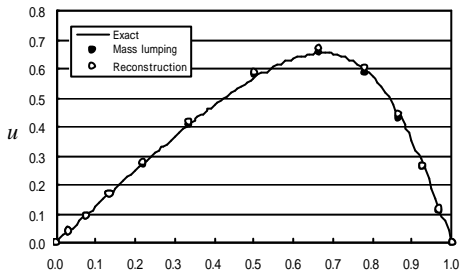
	Case1	Case2	Case3	Case4	Case5	Case6	Case7	Case8
Mesh	Mesh1	Mesh2	Mesh2	Mesh3	Mesh1	Mesh2	Mesh2	Mesh2
Element number	12	12	12	12	24	24	24	24
		2.0	4.0			2.0	4.0	
$x_{min}$	$8.333 \times 10^{-2}$	$3.096 \times 10^{-2}$	$8.841 \times 10^{-3}$	$6.250 \times 10^{-2}$	$4.167 \times 10^{-2}$	$1.419 \times 10^{-2}$	$3.691 \times 10^{-3}$	$3.125 \times 10^{-2}$
$t$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-4}$	$1.0 \times 10^{-5}$	$1.0 \times 10^{-4}$
Time step number	$4.0 \times 10^3$	$4.0 \times 10^3$	$4.0 \times 10^3$	$4.0 \times 10^3$	$4.0 \times 10^3$	$4.0 \times 10^3$	$4.0 \times 10^4$	$4.0 \times 10^3$



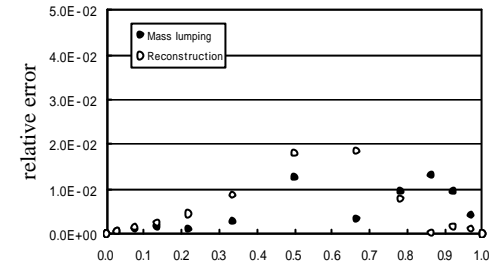
(a) Case 1



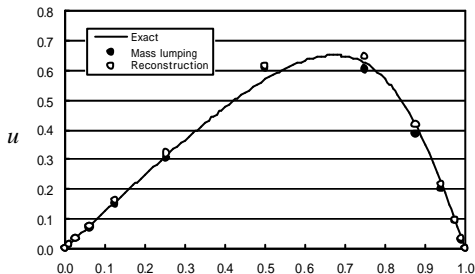
(a) Case 1



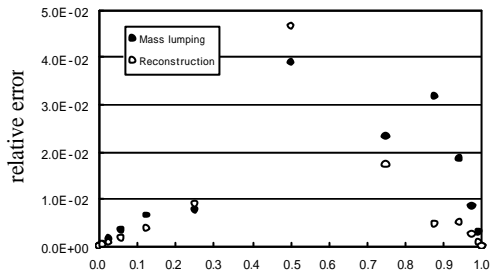
(b) Case 2



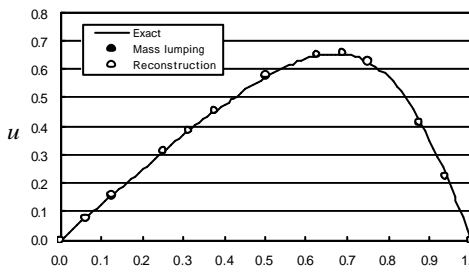
(b) Case 2



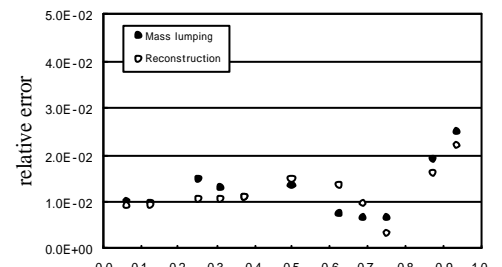
(c) Case 3



(c) Case 3



(d) Case 4



(d) Case 4

Fig. 8 The comparison between mass lumping and data reconstruction

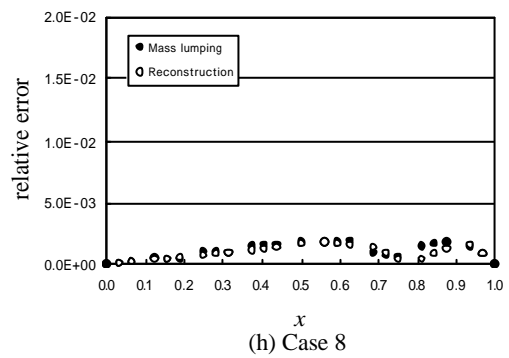
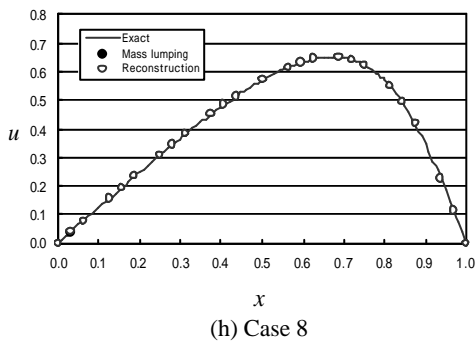
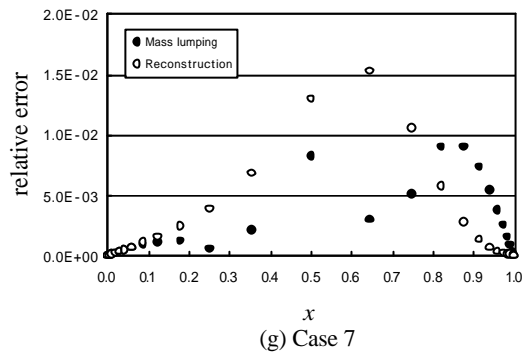
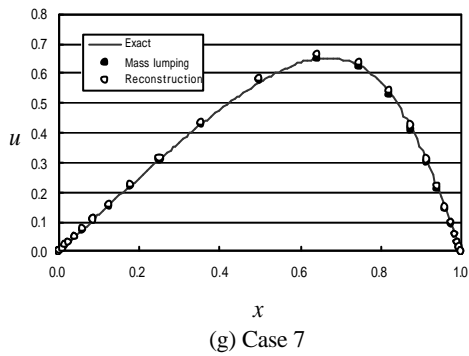
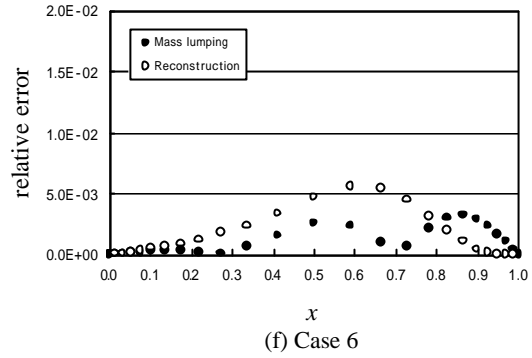
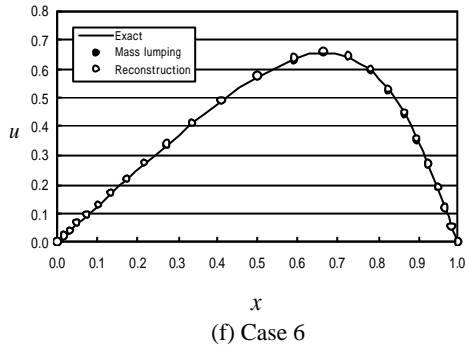
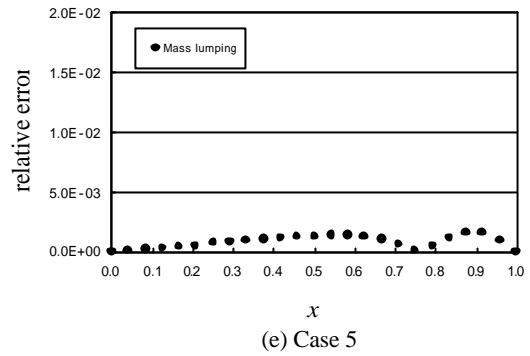
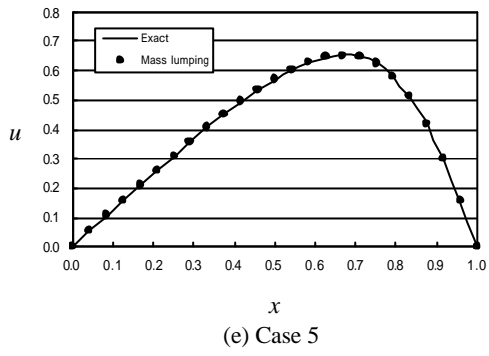


Fig. 9 The comparison between mass lumping and data reconstruction



### (3) GSMAC 有限要素法への導入についての検討

質量の集中化の問題点を解決すると、Navier-Stokes 方程式の計算手法である GSMAC 有限要素法において、解の精度向上や Poisson 方程式の反復回数の減少といった効果を期待できる。1次元ではデータの構築点である重心は3つである。一方、2次元では  $x$  方向の微分値と  $y$  方向の微分値で構築点が異なるのでデータの補正が増え、計算時間が膨大になる。したがって、ある程度まで改善精度を低くして、重心を1つに近似する必要がある。このことについては、さらに検討を進める必要がある。

### 5. 結言

- ・質量集中化の問題点を解決するために、データを再構築する方法の一つを提案した。
- ・有限要素法で離散化した拡散項のデータを重心での値として評価することの妥当性を示すことができた。
- ・1次元の検証問題として拡散方程式、Burgers 方程式を数値解析したが、データの再構築により領域全体の精度を向上させることはできなかった。

### 参考文献

- (1) 棚橋, 流れの有限要素法解析, (1997), 朝倉書店.
- (2) 登坂, 大西, 偏微分方程式の数値シミュレーション, (1993), 東京大学出版会.