

Title	Element-Free Galerkin Methodに適応した並列アルゴリズムの開発
Author(s)	渡邊, 正宏
Citation	
Issue Date	1998-03
Type	Thesis or Dissertation
Text version	author
URL	<a href="http://hdl.handle.net/10119/1141">http://hdl.handle.net/10119/1141</a>
Rights	
Description	Supervisor:松澤 照男, 情報科学研究科, 修士

# Element-Free Galerkin Method に適応した 並列アルゴリズムの開発

渡邊 正宏

北陸先端科学技術大学院大学 情報科学研究科

1998年2月13日

キーワード: メッシュレス, EFGM, FEM, 並列計算, 領域分割.

近年のコンピュータの進歩により、数値解析の分野ではますます大規模な問題を扱うようになってきている。差分法やその他の解析手法も用いられるが、特に複雑な流れ場や構造解析で、有限要素法を用いる事も多い。有限要素法は、複雑な構造物や流れの問題でも、その形状や境界条件さえはつきりしていれば、その挙動を高い精度でシミュレートすることが可能である。さらに、超並列計算機を用いて並列処理を行なう事によって、精度良く高速に解を得られるように、その計算手法やアルゴリズムが研究が進められている。有限要素法は、多くの計算量を必要とするが、計算のほとんどはシステム化されたマトリックス演算であり、コンピュータに適した計算手法である。ただし、解析モデル、メッシュ分割、計算条件などが正しくなければ精度の良い結果は得られない事が有限要素法の欠点である。特に、メッシュ分割の善し悪しが、結果の善し悪しを直接左右する事が知られている。精度の良くなるメッシュの張り方などが研究されているが、実際には経験による部分が大きい。よって、コンピュータの発展により解析の計算速度は向上しているが、計算条件を設定する準備段階に多大な時間が費やされている事になる。今後ますます、問題が大規模かつ複雑化していくことが考えられるため、精度の良いメッシュの準備は大きな問題となる。近年研究されている領域の自動分割手法を用いれば 解析実施者は領域の定義のみを行ない、解析に必要な要素分割や格子分割は自動的に生成される事になる。

その一方で、解析の前準備の複雑さを軽減するために、要素分割を必要としないメッシュレス解法の研究が盛んに行なわれている。Element-Free Galerkin Method(以下 EFGM)は、メッシュレス法の中で有限要素解析である Galerkin 法に整合性よく結び付けることができる。この方法は、その名の通り有限要素解析で準備することに労力を要するメッシュ分割を必要とせず、節点のみを自由に配置することができる。メッシュを設置することが困

難なモデルであっても、節点を配置すれば計算が行なえる事になり、解析の準備に掛かる労力を軽減できる。

EFGM は、一般の有限要素法に比べ内挿関数の計算に時間が多くかかる。それは、積分点周辺の節点を検索する必要があるためである。積分点毎に検索が行なわれるため、節点数が多いほど計算に時間を必要とする。各積分点毎に、その周辺節点の情報を記憶すれば検索時間を浪費しないが、メモリを消費するのであまり得策ではないと考えられる。

本研究では、EFGM の並列化を最終目標として研究を進める。EFGM を並列化すれば、積分点周辺の検索時間、係数行列の作成時間が短縮されることは、容易に想像されるからである。しかし、それ以前に EFGM が持つ問題点として、自由な節点配置と台半径関係が問題である事が実験より分かった。EFGM は台半径と呼ばれる距離によって、積分点周辺の節点を選びだしている。この台半径が大きければ節点は多く含まれることになる。しかし、節点があまりに多く含まれると、解が得られる許容誤差を越えてしまう。そこで、EFGM で台半径を積分点毎に逆に節点数から制御する方法を提案した。この可変な台半径により、固定な台半径と比べて簡単に台半径を決定することができ、許容範囲内で解を得ることが容易となった。

本研究では並列化に至る前に、1次元の非線形問題として Burgers 方程式を、2次元の線形問題として Poisson 方程式を、そして2次元の非圧縮流れの問題として Navier-Stokes 方程式と連続の式を EFGM によって解析した。また、合わせて可変な台半径についても実際に適応しながら検証を進めた。

最終的な目標である、EFGM の並列化についても 2次元非圧縮定常流れを計算する事で、検証を行なった。

EFGM としての並列化は、基底関数の計算の並列化に他ならない。なぜなら、基底関数の計算を除けば、一般的な Galerkin 法との計算上の手続きは違いがないからである。ただし、節点配置が自由である事、台半径にて含まれる節点が変わることが、並列計算を行なう上での問題である。さらに、負荷分散という観点から領域分割が重要な問題になる。領域毎の節点密度に応じた領域分割の方法について考える。