I－310　非線形有限要素解析におけるオブジェクト指向解法

法政大学大学院　学生員　田中 成紀
法政大学工学部　正　員　前田 重行
法政大学計算センター　正　員　武田　洋

[1. はじめに]　この論文は従来の有限要素解析ソフトの問題点と，オブジェクト指向プログラミングの基礎概念を紹介する．非線形有限要素の基礎は，オブジェクト指向数値解析プログラムを用いて説明する．さらにベクトルやマトリックスクラス等を，基本的な代数マトリックスを計算するために用いる．それらのクラスは演算子の多重定義の表記法を用いることによって，明確な構造化プログラムを形成する．そのためオブジェクト指向プログラミングは，複雑で間違い易い典型的な例として知られている非線形有限要素解析プログラムを改良するために用いる事が出来る．この論文では非線形有限要素解析に対して，オブジェクト指向プログラミングを用いることによる，この手法の有効性及び数値解析における効率性の検討をすることが目的である．

[2. オブジェクト指向方法]　C，Pascal や Fortran の様な手続き型言語は，受動的なデータを関数と手続きによって処理する．またデータと算法は異なった構造をしているので，それぞれ異なった方法で宣言や定義を行う．しかしオブジェクト指向は一つのオブジェクトがデータとメソードと呼ばれる関数を含み，オブジェクトはそのメソードがそのデータのみを操作する事ができる．そのため外部関数はこのオブジェクトのデータを変化させることは出来ない．外部関数と他のオブジェクトはメッセージを送ることによってだけ，ある一つのオブジェクトと連結される．一般にメッセージはメソードの呼び出しに使用される．そのためオブジェクトがメッセージ受けると，その手続きの一つを実行する．オブジェクトの構成は図１に示す．

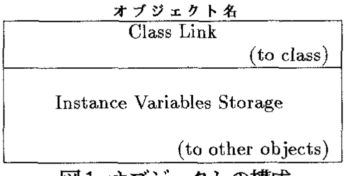

図1 オブジェクトの構成

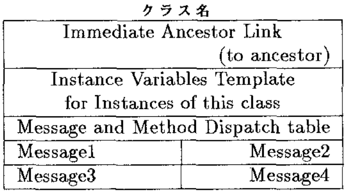

図2 クラスの構成

プログラムはオブジェクト間の幾つかの伝達を含んでいる．同じメッセージを異なった手続きを持つ異なったオブジェクトに送ると，同じメッセージに対しても異なった挙動を示す，この挙動は多重定義と呼ばれている．この多重定義によって一つのオブジェクトの一つの演算子は異なった意味を含んで持つことが可能である．

同質のオブジェクトは一つのクラスの中に含まれる．一般にクラスの宣言はCの構造体の型定義の様な，様々なデータタイプの構造と多くのメソードを含んでいる．クラスはツリー構造と同様な形式の階級組織クラスの中に配列される．上位クラスのデータとメソードは下位クラスに継承される．クラスの構成は図２に示す．

[3. 非線形有限要素解析]　非線形有限要素解析の定式化，組立や適用に対する従来の方法はオブジェクト指向環境に簡単に移行することが出来る．ここでは非線形有限要素法の理論上の基礎とクラスについて記述する．

変位の大小に関わらず，内部及び外部の‘力’の釣り合い条件は満たされなければならない．そして，もし変位が有限個の(節点)パラメータuによって一般の方法で規定されるなら，仮想仕事の原理を用いて必要な釣り合い方程式を得ることが出来る．しかしながら，ここではお互いに関連する‘応力’と‘ひずみ’の異なった定義を用いなければならない．これは一般に次式の様に書くことが出来る．

$$\mathbf{R}(\mathbf{u}) = \int_V \overline{\mathbf{B}}^T \sigma dV - \mathbf{f} = \mathbf{0} \tag{1}$$

$\mathbf{R}$ は外部と内部の一般的な力の総和を表し，そして $\overline{\mathbf{B}}$ はひずみの定義から次式のように定義される．

$$d\varepsilon = \overline{\mathbf{B}} d\mathbf{u} \tag{2}$$

もしひずみが大きければ，バーはひずみが非線形関係で変位に依存するために加えられる，そしてマトリックス $\overline{\mathbf{B}}$ は $\mathbf{u}$ の関数となる．次式の様に表すと便利である．

$$\overline{\mathbf{B}} = \mathbf{B}_O + \mathbf{B}_L(\mathbf{u}) \tag{3}$$

$\mathbf{B}_O$ は線形微小ひずみ解析におけるマトリックスと同じである，そして $\mathbf{B}_L$ は変位の関数である．一般に $\mathbf{B}_L$ はこの様な変位の関数であることがわかる．もしひずみが微小ならば，一般に弾性関係は次式のように書くことが出来る．

$$\sigma = \mathbf{D}(\varepsilon - \varepsilon_O) + \sigma_O \tag{4}$$

$\mathbf{D}$ は普通の弾性マトリックスである．

もしニュートン・ラプソン法を用いるならば，ここで説明しているような $d\mathbf{u}$ と $d\mathbf{R}$ 間の関係を必要とする．この様に $d\mathbf{u}$ 関して公式(1)の変分をとれば次式を得る事が出来る．

$$d\mathbf{R} = \int_V d\overline{\mathbf{B}}^T \sigma dV + \int_V \overline{\mathbf{B}}^T d\sigma dV = \mathbf{K}_T d\mathbf{u} \tag{5}$$

公式(4)と(2)を用いると次式のようになる．

$$d\sigma = \mathbf{D}d\varepsilon = \mathbf{D}\overline{\mathbf{B}}d\mathbf{u} \tag{6}$$

もし公式(3)が成り立つならなら次式となる．

$$d\overline{\mathbf{B}} = d\mathbf{B}_L \tag{7}$$

それ故に次式を得る．

$$d\mathbf{R} = \int_V d\mathbf{B}_L^T \sigma dV + \overline{\mathbf{K}}d\mathbf{u} \tag{8}$$

$\overline{\mathbf{K}}$は次式に表す．

$$\overline{\mathbf{K}} = \int_V \overline{\mathbf{B}}^T \mathbf{D}\overline{\mathbf{B}}dV = \mathbf{K}_O + \mathbf{K}_L \tag{9}$$

$\mathbf{K}_O$は一般の微小変位剛性マトリックスを示す，即ち次式となる．

$$\mathbf{K}_O = \int_V \mathbf{B}_O^T \mathbf{D}\mathbf{B}_O dV \tag{10}$$

マトリックス$\mathbf{K}_L$は大変位によるもので，それは次式によって与える．

$$\mathbf{K}_L = \int_V (\mathbf{B}_O^T \mathbf{D}\mathbf{B}_L + \mathbf{B}_L^T \mathbf{D}\mathbf{B}_L + \mathbf{B}_L^T \mathbf{D}\mathbf{B}_O) dV \tag{11}$$

$\mathbf{K}_L$は初期変位マトリックス,大変位マトリックス等としてよく知られている，そして$\mathbf{u}$の中に線形と２次のある条件だけを含む．公式(8)の第１項は一般に次式のように書かれる．

$$\int_V d\mathbf{B}_L^T \sigma dV = \mathbf{K}_\sigma d\mathbf{u} \tag{12}$$

$\mathbf{K}_\sigma$は応力レベルに関する対称マトリックスである．このマトリックスは初期応力マトリックスや幾何剛性マトリックスとして一般によく知られている．以上の結果から次式を導く．

$$d\mathbf{R} = (\mathbf{K}_O + \mathbf{K}_\sigma + \mathbf{K}_L)d\mathbf{u} = \mathbf{K}_T d\mathbf{u} \tag{13}$$

この様に$\mathbf{K}_T$は全体の接線剛性マトリックスである．

この前述の全式はオブジェクトとして考えられるクラスを含んでいる．力，変位，ひずみと応力はクラスを用いて定義する．この理論での重要な概念は空間内での参考点，即ち節点の考え方である．力，変位，境界条件，そして幾何学的領域でさえ節点に依存する．有限要素の記述を始めるためのクラスは節点のクラスである．節点の固定は自由度と関係づけられた変位を固定することで表す．

数値積分有限要素の計算の利点はいくつかの要素タイプに対して形状関数と導関数を定式化する事によって示す事が出来る．一般的問題に対する数値積分公式はGauss Legendreの求積法に基づいて与える．この概念は一般的な要素クラスの設計の中に利用される．

[4. 組立] 節点，材料，要素の定義及び有限要素定式化の基本的な構成を含むクラスライブラリーを得ると，次にそれらを有益なプログラムに組み上げることが必要となる．効率よいオブジェクト指向プログラミングはオブジェクトの適切な管理と相互の関係が重要である．そこで線形リストを用いる事によるオブジェクトの組織化に関する標準的な方法を採用する．線形リスト管理はオブジェクトのリストを扱う一般的なクラスを作ることによって行う．このクラスは‘オブジェクトリスト’と呼ぶ．オブジェクトリストクラスの例は節点インスタンスの組織化を用いる．このリストは節点を扱う幾つかの手続きを持ったオブジェクトリストである．これは全てのリストの一般的な方法を含む．

高度に洗練された有限要素解析プログラムは答えを計算するために複雑なデータと制御構造を用いる．オブジェクト指向プログラミングは抽象的なレベルに作用するクラス階級組織を作ることによって下位レベルのプログラム記述を簡単にする．基本のクラスは実際の数に作用する，ところが制御と組織的な関数を実行するクラスは形成される抽象的なデータ形式を処理する．

オブジェクト指向言語の最も重要な特徴の一つは現存するソフトを利用する能力である．ベクトルとマトリックスクラスを含む幾つかのクラスは基本的な代数マトリックスを操作するために導入される．

[5. おわりに] 一般的な結論として，オブジェクト指向プログラミングは非線形有限要素解析プログラムを改善するのに有益であるといえる．オブジェクト指向法を用いたプログラミングはより少ない時間で行え，より小さいプログラムになり，同等の手続き言語よりもデータと手続きの管理しやすい環境を提供する事が出来る．

参考文献 1)S.Maeda,S.Tanaka and H.Takeda,An object oriented-approach for non-liniar finite element anarisys *Non-liniar anarisys and design for shell and spatial structures.*,105-112 (1993). 2)S.P. Scholz, Elements of an object-oriented fem++ program in c++. *Computer and Structure.*, **Vol.43,No.3**, 517-529 (1992). 3)B.W.R. Forde, R.O. Foschi and SF. Stiemer, Object-oriented finite element analysis. *Computer and Structure.*, **Vol.34,No.3**,355-374 (1990).