

構造計算における電子計算機の役割

総括執筆(座長) 大 地 羊 三*
 話題提供者 飯 田 隆 一**
 同 宮 田 尚 彦***
 同 大 坂 憲 司****

1. 構造解析における数値計算の位置づけ

話題提供者の一人飯田は図-1を示し、次のように説明している。電子計算機が登場してしばらくの間は、以前から使われていた数値計算の手法をプログラム化することが常識であった。しかし、これらは必ずしも電子計算機に適した方法とはいえず、また、理論式の誘導にあたって加えられる制約が多いため、適用範囲が狭いといった欠点があった。しかし近年は、しだいに電子計算機に適した計算法が開発されるようになってきている。その多くはマトリックスの手法が使われており、できるだけ早い段階で理論から離れ、適用範囲を広げる方向が取られている。また大阪は、物理現象を解析し、数値計算に結

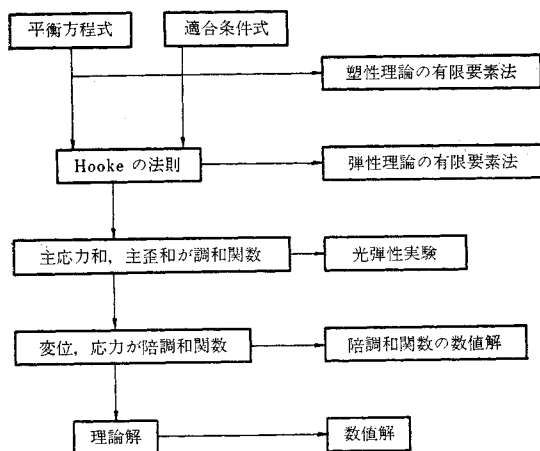


図-1 弾性解析における数値計算の位置づけ (飯田)

* 正会員 工博 法政大学教授 工学部土木工学科
 ** 正会員 建設省土木研究所ダム部 フィルダム研究室長
 *** 正会員 国鉄構造物設計事務所 主任技師
 **** 正会員 三菱重工(株) 技術部技術管理部

びつけて結論を得るまでの流れを 図-2, 3 にまとめています。図中点線の部分が新しい計算法の指向している方向とみてもよいであろう。さらに宮田は、計算法に質的向上があったとし、以前は略算法に頼らざるを得なかった、複雑な形状をもつ構造物の部材軸方向変位量の計算ができるようになったことをあげている。

以上のように、電子計算機が登場により、計算法が質的に変化し向上したことは認められるが、一方において誤差の問題が大きく浮び上がってきている。

2. モデル化による誤差の問題

図-3 に構造物をモデル化し、数値解析モデルをつくるまでに導入される誤差が列記されている。いかなる構造物でも、これをそのままの形で解析することは困難である。数値計算ができる形にモデル化しなければならない。モデル化による誤差は、電子計算機特有のものではない。しかし、電子計算機を駆使して複雑な構造計算ができるようになればなるほど、モデル化の誤差がクローズアップされる。

飯田・宮田が指摘するように、弾性定数ひとつ取っても、鉄筋コンクリートや土の場合にこれらをどのように選ぶか問題である。もし、ひずみまたは応力の関数にしなければならないものならば、そのような計算ができるようになっただけに、定数になる場合その根拠が問われるわけである。複合応力の塑性条件等も再検討の要がある。さらに、実際的な問題として、たとえば格子桁のねじり剛性は無視してもよいものであろうか。たわみ角法で無視されていた部材軸方向の伸びは、変形法で改良され、その誤差の程度が解明されている。格子桁のねじり剛性の無視は、ラーメンの軸方向伸びの無視とは逆の意味を持つものであるが、その誤差の程度を検討しておく

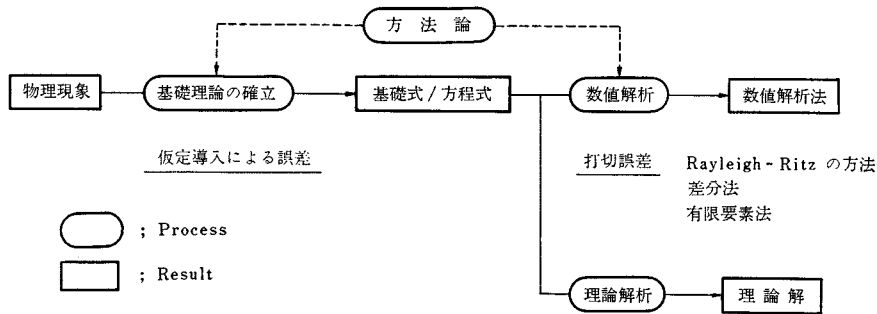


図-2 物理現象の解析と数値計算 (大阪)

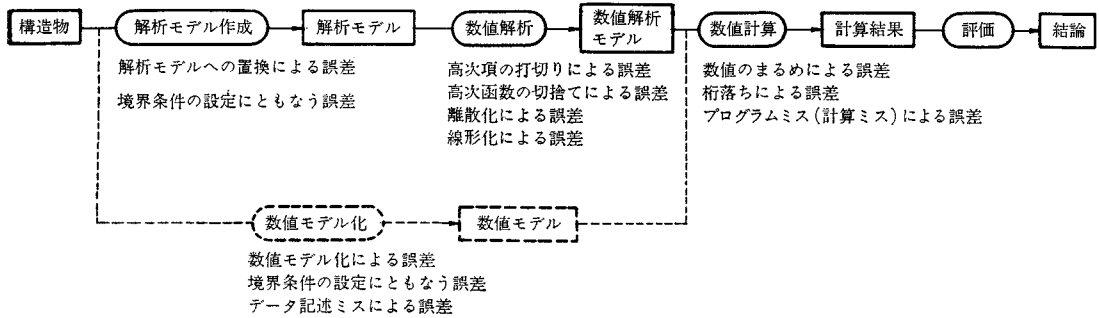


図-3 構造解析と数値計算 (大阪)

必要がある。

また、立体トラスをトラス構造物として解いてよいものであろうか。弦材が節点で連続している場合が多いだけに一考を要する。立体構造物が解けるようになった現在、昔流のモデル化だけにとらわれるべきではない。

地中の土圧分布を計算する場合のように、無限に広がる対象物をそのままの形で計算することはできない。有限の位置に仮定の境界を設定する必要がある。この仮定の境界に与えた境界条件による誤差も、モデル化の誤差のひとつである。

以上のほかに、打ち切り誤差といわれる一連の誤差がある。たわみなどを級数で近似した場合、高次項を打ち切ったための誤差などがこの例である。非線形な挙動を示す物理量を、計算を簡単にする目的で、線形化するために導入される誤差もこの分類に入れてよいであろう。鉄筋コンクリートや土の応力-ひずみ曲線を直線で近似する場合等を思い浮かべていただきたい。

電子計算機は、連続関数をそのままの形で表現することが不得意である。したがって、いくつかの点で関数値が計算され、これらがそのあとの計算の基礎になる。電子計算機では、このような離散化のための誤差はさけることができない。

以上のように、構造物をモデル化する際に各種各様の誤差が入り、計算結果の精度を悪くしている。各種の実験結果をよく検討し、計算の目的に合ったモデルをつくり、これを受け入れることができるプログラムを選定し

て構造計算を行なうべきである。近年、構造計算用のプログラムは Black Box 化され、データさえ与えれば計算結果が得られるようになったが、計算目的に合ったモデルを作る作業および、これを計算させるプログラムを選定する自由は人間の側にある。

最後に大阪が提示した、有限要素法による片持ばりのモデル化の例を図-4、5 および表-1 に示しておく。

図-4 は計算に使用した三角形要素の説明図であり、(b) は要素内の変形を一次式で近似し、3 節点で他の要素と接続するようにしたもの、(a) は要素内の変形を 2 次式で近似し 6 節点で他の要素と接続するようにしたものである。これらの要素を

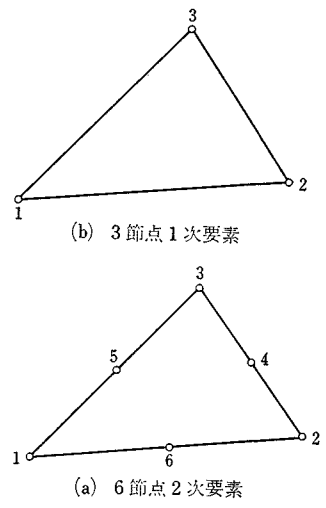
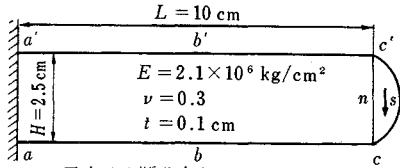
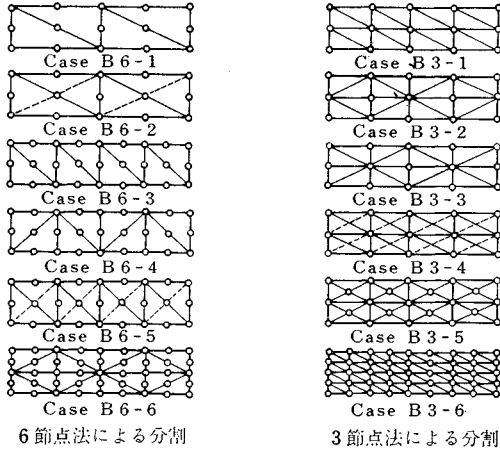


図-4 有限要素

を用い、図-5 の最上段に示す片持ばりをその下に示すようにモデル化し、自由端に $S=600 \text{ kg}$ の荷重をかけて計算したときの自由端のたわみが表-1 である。表-1 から、本例題の場合、3 節点一次要素 (B3) が全く無力であることがわかる。十分な分割がなされていると思わ



二次せん断分布荷重 $S = 600 \text{ kg}$



(大阪)

図-5 片持ばりの要素分割

表-1 有限要素法による片持ばりの計算結果 (大阪)

| モデル番号 | 節点数 | 自由端のたわみ | たわみ比 | モデル番号 | 節点数 | 自由端のたわみ | たわみ比 |
|-------|-----|------------------|----------------|-------|-----|------------------|----------------|
| B6-1 | 15 | 0.719 | 0.94 | B3-1 | 15 | 0.293 | 0.38 |
| B6-2 | 15 | 0.715 | 0.94 | B3-2 | 15 | 0.305 | 0.40 |
| B6-3 | 27 | 0.745 | 0.98 | B3-3 | 15 | 0.324 | 0.42 |
| B6-4 | 27 | 0.744 | 0.98 | B3-4 | 15 | 0.291 | 0.38 |
| B6-5 | 27 | 0.743 | 0.97 | B3-5 | 23 | 0.481 | 0.63 |
| B6-6 | 45 | 0.752 | 0.99 | B3-6 | 45 | 0.535 | 0.70 |
| 理論値 | | 0.763 (0.732) | 1.00 (0.96) | 理論値 | | 0.763 (0.732) | 1.00 (0.96) |

注：() 内は曲げによるもの。理論値は計算仮定に対して必ずしも厳密ではない。

れモデル B3-6 の場合ですら、大きな誤差を含んでいることに注意されたい。

3. 数値計算上の誤差の問題

構造物のモデル化ができ、数値計算に入った段階でも、いくつかの誤差が予想される。これらはプログラミング時の誤差と入力データ作成時の誤差に分けると都合がよい。前者は桁落ちの誤差と丸めの誤差に代表される。指数型で数字を表現することにすれば、電子計算機内では、非常に大きな数から非常小きな数まで取扱うことができる。しかし、有効数字には限度があり、これが誤差の発生源になる。

たとえば、有効数字 4 桁で $\cosh(5.0)$ と $\sinh(5.0)$ を計算すると、それぞれ 0.7421×10^2 お

よび 0.7420×10^2 となるが、これらから $\cosh(5.0) - \sinh(5.0)$ をつくると、 0.1×10^{-1} となり、有効数字は 1 桁に落ちてしまう。このような困難な事態をさけるためには、双曲線関数を次式で指数関数になおし、

$$\cosh x - \sinh x = \frac{1}{2}(e^x + e^{-x}) - \frac{1}{2}(e^x - e^{-x}) = e^{-x}$$

$e^{-5.0}$ を有効数字 4 桁で求めて 0.6734×10^{-2} としなければならない。 x の値が小さい所で $(1 - \cos x)$, $(x - \sin x)$ などを計算する場合にも同じような問題が起る。

この種の誤差は、使用する計算機にも左右される。たとえば、16 ビット×2 が単位の IBM の電子計算機と 64 ビットが単位の CDC の電子計算機では、誤差の出方に差があることは容易に想像できるであろう。

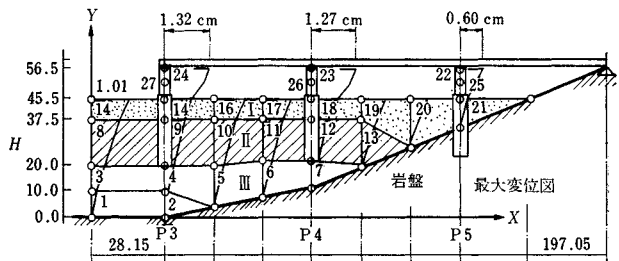
有効数字の桁数に制限があるための、いまひとつの誤差は丸めの誤差である。手計算の場合にたかだか連続して 100 回程度の四則演算であろうから、丸めの誤差で有効数字が 1~2 桁落ちる程度ですむ。電子計算機では、数万回の四則演算もまれではない。したがって、丸めの誤差が有効数字に与える影響も手計算の比ではない。この例として、宮田は図-6 に示す橋梁モデルの地震応答解析の結果をあげている。エル・セントロの地震波を入力にして 1/100 秒刻みで応答解析を行なったが解が得られず、1/400 秒刻みで計算せざるを得なかったことである。この例では、丸めの誤差ばかりではなく、微分方程式特有の誤差の伝播といった問題も含まれているのでやっかいであるが、計算の途中で誤差の推定ができる解法を採用すべきであろう。

また大阪は、構造計算でよく用いられる連立一次方程式の解を求める問題において、演算量が極端に変わること、次の例で説明している。図-7 は仮想の構造物であり、(A), (B) 2 種類の節点番号のつけ方が示されている。これらに対する記憶容量と演算量は表-2 のようになる。

表-2 節点番号のつけ方による差 (大阪)

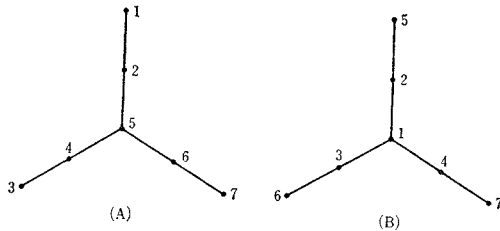
| | (A) | (B) | (C) |
|--------|-----|-----|-----|
| 所要記憶容量 | 13 | 22 | 28 |
| 被変換回数 | 12 | 43 | 77 |

表の (A), (B) は図-7 の (A), (B) に対応するものであり、いづれも非零要素だけを電子計算



(宮田)

図-6 奥羽本線米代川橋梁の地震応答解析結果



(大阪)
図-7 構造物の節点のつけ方

機内に格納する場合である。(C)は零要素も電子計算機内に格納されており、非零要素、零要素の区別をつけないで計算を進めた場合を示す。表中の数字は、実際の記憶容量あるいは演算量を示すものではない。記憶容量は1節1点自由度とみなしたときのものであるから、平面構造物ならば $3 \times 3 = 9$ 倍、立体構造物ならば $6 \times 6 = 36$ 倍する必要がある。また、被変換回数は、(3,3)あるいは(6,6)のマトリックスに対してなされた一連の演算を1回と勘定している。したがって、相対的な演算量としてみなければならないが、それにしても(A)のモデルが(B),(C)に比べて極端に有利であることがわかる。

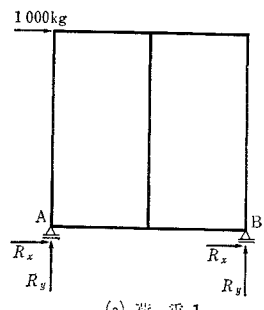
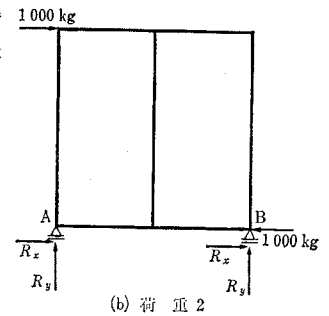
演算量を減らすことは丸めの誤差を少なくすることにつながると同時に、計算時間を減らす効果もある。大型の連立一次方程式を解くことをきらい、方程式の未知数を減らす方向に力点を置くことがよく行なわれているが一般に未知数を減らすためには、構造形式に制限を加えたり、方程式の係数の計算に複雑な演算が必要になる場合が多く、これをプログラム化するとその適用範囲がせばめられ、同時に演算量が増加するということが結局経済的にひき合わなくなることが少なくない。

プログラムミスについても一言する必要がある。この種のミスは、プログラムが完成した時点で完全に排除されていなければならないことは論をまたない。しかし、複雑なプログラムになると、そのすべての部分がチェックできるようなチェックデータをつくるのが困難であるし、もしできたとしても、それを使って計算した結果が正解であるということを判定することがむづかしい。このような理由もあって、簡単な例題だけでチェックされたプログラムが実用に供されている場合も多いように見受けられる。このような傾向は決して許されるべきものではないが、現状では利用者の側で自衛する以外に手がないうである。あるいは、公平な第三者によるチェック機関といったものが考えられるかも知れない。

入力データ作成時に導入される誤差としては、入力データそのものが不備であるための誤差と、利用しようとしているプログラムで計算可能な範囲を越えた入力データを使用したための誤差が考えられる。大阪は、次の2例でこの問題を説明している。その第一は、境界条件の設定を誤り、結果として図-8に示す構造物を解いた場

合であり、解として得られた支点反力は、表-3のようになった。支点反力は作用外力とつり合っておらず、明らかに誤りである。このような簡単な問題であれば誤りはただちに発見できるであろう。しかし、複雑な構造物になると、計算結果をチェックすること自体が大変な作業になる。プログラムの中にチェックルーチンを内蔵させることが望ましい。

第二は、計算可能な限界を無視した断面を仮定したための誤差累積の例である。図-9



(大阪)
図-8 境界条件設定誤りの例

に示す構造物で、太線の部分と細線の部分の曲げ剛性の比(κ)を $1.0 \sim 10^7$ に変化させた場合の水平および垂直反力の総和が表-4に示されている。 κ が $10^3 \sim 10^4$ 以

表-3 境界条件設定の誤りによる計算結果 (大阪)

| 支持点 | 荷重-1 | | 荷重-2 | |
|-----|---------|---------|--------|---------|
| | R_x | R_y | R_x | R_y |
| A | -303.22 | -793.18 | 0.0055 | -999.98 |
| B | -159.41 | 793.33 | 0.0052 | 999.98 |
| 計 | -462.63 | 0.15 | 0.0107 | 0.00 |

注: UNIBAC 1108, 8桁で計算したものの。

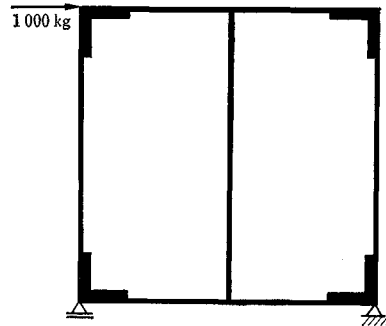


図-9 数値計算の限界を無視した断面仮定の例 (大阪)

表-4 数値計算の限界を無視した断面仮定による計算結果 (大阪)

| κ | 水平反力総和 | 垂直反力総和 | κ | 水平反力総和 | 垂直反力総和 |
|----------|-----------|--------|----------|-----------|----------|
| 1.0 | -999.8836 | 0.0064 | 10^4 | -987.5589 | 1.8970 |
| 10 | -999.8963 | 0.0214 | 10^5 | -868.2582 | 28.1668 |
| 10^2 | -999.7910 | 0.0190 | 10^6 | -624.1831 | 74.5414 |
| 10^3 | -998.0396 | 0.1660 | 10^7 | -6.6512 | -76.5802 |

注: UNIBAC 1108, 8桁で計算。

上になると急に精度が悪くなるのがうかがえる。プログラミングの技術で ϵ の限界値を上げることは可能であるが、それにも限度があろう。

一つの問題の解析解が与えられている場合に、断面二次モーメント、断面積、場合によっては長さまでも、無限大にしたり0にしたりして、極限の状態にある問題の解を求めようとする方法がよく行なわれている。電子計算機による数値に、この方法を無差別に適用することはできない。これを可能にするためには、プログラム上で特定な数以上を無限大と判定したり、割算の前で分母が0であるか否かの判定をするなどの、きめ細かな配慮が必要である。

4. 今後に望みたい方向

以上のように、電子計算機によって構造計算をする場合、その計算過程の各段階で誤差の発生が予想されるがこれらのすべてが解決されて初めて正確な構造計算が可能になるわけである。このうち、プログラム自体に関係する部分は、それをつくる人たちのプログラミング技術の向上に期待するものとして、利用者の立場から飯田・宮田は人出力データに起因する誤差をなくす方策として① 入出力データの改善と簡素化、② 入出力データをチェックするシステムの整備、③ 入出力用共通記号、言語の開発、④ プログラム使用説明書の完備等の提案があった。これらの中にはプログラム作成者の努力で改善できるものもあるが、利用者側の強力な呼びかけを必要とするものが多い。むしろ境界の問題として両者の共同開発が望まれる。

また、設計を対象にした場合、データを入れれば計算の最終結果がプリントされるといった一貫した自動設計用のプログラムではなく、計算の途中結果がプリントされ、そこで人間の判断が入られる対話型のプログラムが要求される。このためには、① 情報の分類、整理、検索(データプロセス)のためのシステム、② 概略計算が可能なプログラム、③ 不要部分または訂正部分がただちに省略あるいは改善できるプログラム、④ 簡単な図形読込みの方法などの開発が必要であるとしている。さらに、電子計算機の特性を生かした新しい計算法、たとえば、シュミレーション、統計解析、数値計画法などの利用も必要であろう。

電子計算機による構造計算の将来を予測する欧米の論文の中に必ず出てくる Problem Oriented Language (POL), Man to Machine といった考え方がうかがえる点で、わが国でも欧米でも将来を同じように予測しているように思われて興味深い。ただ、Optimum Design といった言葉が表面に出なかった特別な理由があるか否

かは判明しない。

最後に、宮田は電子計算機導入による弊害として、教育の問題を取り上げ、次のように論じている。これは、あとで討論の主題になったものであるから、その全文を載せることにする。すなわち

「構造解析、設計計算が手計算で行なわれていた時代には、設計者は構造物の設計の過程において、思考し判断することができたため、設計が完了した時点において創造の喜びがあった。しかし、電子計算機を用いた計算の過程には、人間の思考判断の介入する余地が存在しないため、定量感、充足感が持てないのは事実である。また、そのうえ電子計算機に対する依頼心がふえれば、その解析結果に対する盲信をひき起すことになり、しだいに判断する心は育たなくなるものと思われる。これをさけるには、構造解析、断面計算の過程を数多くの例でとらえ、手計算により、あるいは既往の資料により、判断を行なう心がけが必要である。構造解析の前提になる設計計画を例にとれば、これを行なう設計者には、その人の既往の実績と知恵が要求される。この知恵は、電子計算機に頼りきっていて得られるであろうか。

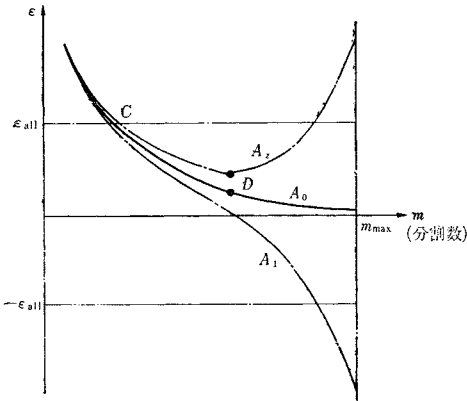
電子計算機を構造計算に利用することによって、たしかに設計者の労苦は解放された。しかし、入力データを与えると電子計算機が計算し、その解析結果が得られ、その過程は往々にして体得されないままとなる。計算機を使用するものは、プログラムが作成された本筋を一つ一つ理解しないままで果してよいのであろうか。これらの問題は、今後の構造技術者育成上の教育問題として残されているように思われる。」

5. 討論の経過(その1・教育の問題)

以上が話題提供者の論旨の概要に筆者の意見を加えたものである。討論に入って最初に小松(大阪大学)は、電子計算機を上手に活用して目的を達成するためには、力学と数学の総合的能力の向上が最も大切である、と前置きして、その理由を次のように解説した。

① 誤差には離散化や線形化による誤差(打切り誤差)と四則演算の誤差(丸めの誤差)があるが、これらはいずれも反した性質を持つ。たとえば、有限要素法における要素の大きさあるいは差分法における区間の長さを細分化すればするほど、前者は減少するが後者は逆に増加し、累積誤差は図-10のようになる。したがって、単に細分割するだけでは正解へのアプローチは必ずしも成功するとはいえない。誤差を最小にするためには、各種誤差の力学的および数学的評価手段の導入が大切になる。

② 有限要素法などでよく実行されているように、理論的に厳密な解析解が知られている比較的簡単な構造モ



(小松)
 A_0 : 離散化のみによる誤差
 A_1 または A_2 : 離散化と四則計算による誤差
 ϵ_{all} : 許容誤差

図-10 離散化(または線形化)と四則計算による誤差

デルについて計算し、その結果から使用しようとする計算法の合理性、収束性、誤差の程度を総括的に判断しようということがしばしば行なわれている。この構造モデル(モデルゲージ)の選定を誤ると、せっかくの総合的誤差の評価法の長所が発揮できないばかりでなく、危険な結果を招くことになる。対象にする構造物が複雑になればなるほど、モデルゲージの適切な選定は高度な力学的能力に負うところが大きい。

③ 級数計算や逐次計算で収束の判定に N 回目と $N+1$ 回目の繰返し計算の相対誤差が用いられることがよくあるが、問題によっては大変な誤りをおかす原因になる。たとえば、 $S_n = \sum (1/n^k)$ は、 k が 1 に近づくと収束が非常に悪くなる級数であるが、 $k=1.5$ の場合には

$$S_{21} - S_{10240} \approx 0.2, (S_{20} - S_{21})/S_{21} \approx 0.0047$$

となる。もし、相対誤差 0.5% 以下で打切ることになると、21 項で十分ということになるが、まだ、非常に大きな誤差が残っていることがわかる。この場合にも、誤差に対する数学的評価法の重要性が認められる。

④ 古典的理論解の中には、力学的現象を最も端的に表現できるパラメーターが自然に含まれているので、パラメトリック解析による公式化が容易である。したがって、技術者はパラメーターを基準にして部材の形状寸法を算定することができた。これに対して、電子計算機による構造計算では、簡単な寸法、材料特性が単に原始量として含まれることが多いので、パラメトリック解析の際のパラメーターの選定に非常に高度な力学的考慮が必要になる。また、発言の順序は逆になるが成岡(名古屋大学)から、昔から先輩たちが積上げてきた構造解析に関する成果・実績に眼を向け、これを学習することにより力学の力がつけられる、という主旨の意見があった。

これらに対して竹田(CRC)、川原(中央大学)から

① 誤差については、電子計算機で処理できる数は離散型

の実数であり、数学で扱われる連続型の実数とは本質的に異なる。この点を配慮した新しい誤差論をつくり上げるべきである。また、誤差は構造モデルに依存する度合いが強い。モデル作成にあたって打切り誤差が少なくなるように細心の注意が必要である。② 教育問題については現在の構造力学の解析手段よりは、力学そのものの基礎を身につけるべきであり、電子計算機を前提にした力学が必要である。また、電子計算機を利用すれば繰返し計算が楽にできる。この特徴を最大限に利用して、量的に多くの情報を取れば、それから構造特性をつかむことが可能である、といった発言があった。

一方、倉西(東北大学)は、構造計算には設計計算と解析計算があると思われるとして、次のように述べている。われわれが構造物を設計する場合、最初にある構造物が想定され、いわゆる設計計算が行なわれる。この結果をみて、もとの構造物が改善され、でき上がったものが解析されて、さらにより良いものへと発展していく。この場合、最初に想定される構造物は、人類が長年かけて改良を加えてきたものである。しかし、もし全く新しい物が要求され、その構造物の基本的形式を選び出さなければならぬとき電子計算機は無力ではなからうか。

さらに、小堀(金沢大学)からは、次に示すような具体的な教育計画の提案があった。まず、すでに社会(とくにメーカー)に出た技術者のためには、電子計算機の出力は最終値のみに限らず、従来の設計計算書に近い密度で途中の計算結果を打ち出すようにする。この場合、印刷される用語は技術者が日常使用している用語に合わせ、多少の設計条件の変化は手で計算できるようにする。また、これと平行して新しい電子計算機用設計計算書の様式の規格化を進める。この規格化にあたっては、設計荷重などを単に示方書によるとししないで、入力データの間違いがチェックできるものであること、若い技術者が読んで計算方法過程が理解できる程度のものであることなどが配慮されなければならない。

一方、大学その他の教育機関では、従来の設計計算法を理解させ、構造物各部の寸法等のオーダーを知らしめること、計算機の効用について理解させること(電子計算機は単に数値計算のための道具ではなく、最適設計、シュミレーション、リアル・タイム・コントロールなどにも利用できるといったこと)、計算機用の計算方法や設計方法を開発すること、および社会で開発された電子計算機用設計計算書の読み方、内容、使用法を解説することなどに力点を置く。

以上の討議の内容から、電子計算機が出現する以前に構造力学の勉強をすませ、その真髄をつかまれた諸先生と、学生時代に電子計算機を使いながら構造力学の勉強をし、その後も比較的自由に電子計算機が利用できる立

場にある人たちとでは、考え方に微妙な相違のあることがわかる。諸先生は自分で築かれた構造解析の力で電子計算機による計算結果を解釈されようとしており、構造解析の力をつけることと電子計算機はかかわりがないと判断しておられるのに対して、若い人たちは、電子計算機を道具にして構造解析を行なおうとしており、これを駆使することによって構造解析の力をつけることができると考えているように思われる。構造解析の力のつけ方に定まった方式はないであろう。時代により、立場によって異なっても不思議はないし、個人差も考えられる。要は、宮田がおそれている「電子計算に対する依頼心が増し、その解析結果に対する盲信をひき起す」ことのないようにいましめ、方法論は別にして、構造解析の力をつけることではなかろうか。

教育の問題に対して、話題提供者の考え方はどうであろうか。この設問に対し、大阪は計算内容をくわしく印刷し、これを検討することにより構造解析の力がつけられるとし、飯田は電子計算機の時代に合ったトレーニングをすべきであり、できるだけ数多く計算させるとよいとしている。これに続いて川原から、プログラミングについても同じことがいえる。完成された問題でもよいからいま一度プログラムさせると、プログラムすることによって内容が理解できる、といった発言もあった。

6. 討論の結過（その2・プログラムの共同利用）

このまま討論を進めればさらに多くの人たちから貴重な意見がいただけるようにも思えたが、発言が一瞬とぎれたし時間の関係もあるので、ここで話題を変えプログラムの共同利用について意見を求めることにした。

これに対して、まず飯田裕（建設省）から次のような発言があった。すでにでき上がっているプログラムを重複してプログラムさせるという意見には賛成しかねる。現在多数のプログラムが作成されているし、また作成中のも多い。その中で実用になるものがどれだけあるか疑問である。使用者の立場からは、プログラムは内容を理解すればよいものであって、改めてプログラミングする必要はない。完成したプログラムはくわしい使用説明書をつけて共同利用すべきである。ただし、作成に要する費用等で解決しなければならない問題も多い。

さきの川原の発言は、トレーニングのためにすでにあるプログラムをいま一度プログラムさせるものと理解したい。もし実用に供する目的で力学的に未熟なものにプログラムさせるということであるならば、筆者も賛成しかねる。でき上がったプログラムの効力は、それを作成した人あるいは指導した人の力学的能力に比例するものと考えられる。よく例にあげられることであるが、プロ

グラミングの力は同じであるとして、将棋三級の人がつくったプログラムは三級の力しかなく、三段の人がつくったプログラムは三段の力がある。

次に成岡からに共同開発の機関としてイギリス GEN-ESYS というものがあるが、現在その機構、運営方法などについて調査中であるから判明しだいしかるべき機関に報告する、という発言があった。また、林(大阪大学)からは、全国の大学にある大型計算機センターで開発されたプログラムを登録し、その精度を確認したものについてはセンター間で交換してはどうか、との提案がなされた。まことに結構なことであるが、筆者が意図した共同利用は、業界で重複して開発されているプログラムのむだをはぶくことであった。実用になるプログラム1個を完成するだけでも多額の費用がかかる。まして、すべてのプログラムを1社で開発するということは、大変な投資になるはずである。したがって、プログラム作成者の技術的良心に反して、未完成のプログラムが実用に供される場合も多い。飯田(裕)が指摘したのもこの点ではなかろうか。放置すべきことではない。なんらかの手段を講じて、未完成のプログラムが世に出ないようにすべきであるが、そのみかえりとして、完成されたプログラムの共同利用が考えられるわけである。

大学で開発されるプログラムは、そのほとんどが研究用のものであろう。専門分野の異なる部分で、他大学で開発されたプログラムの利用も十分考えられるが、原則として、それぞれが独自のプログラムになるのではなかろうか。筆者の経験では、専門ではないと思われるプログラミング技術そのものが、でき上がったプログラムの死命を制することもまれではない。大学で研究され開発されたプログラムが、業界の共同利用に供される形こそ理想の姿ではなかろうか。

7. 結 言

討論会に先立って思想統一をしたこともあって、話題提供者の話は、構造計算における誤差の恐しさということに焦点がしぼられていた。誤差に対する理解なしに電子計算機の功罪は論ぜられないと判断したからである。しかし、討論は宮田の発言中にあった教育の問題が中心になり、活発な意見の交換があったことは、まことにご同慶の至りである。司会が要領を得ないため、全員のご意見が聞けず、また結論も出せず申訳なかったが、考えようによっては、統一見解を出さなければならない性格のものではなく、また、出せないことかも知れないと思ひ自己満足している。また、プログラムの共同利用については、利用者の皆様に関心の高いものの一つと考えられるが、核心に入る前に時間ぎれとなり残念であった。