

## コンピュータ・シミュレーションとモデリング ー機能指向からオブジェクト指向へー

知能システム工学領域

坪井一洋

(ksuboi@mx.ibaraki.ac.jp)

### 1. はじめに

シミュレーションという言葉はもともと「見せかけ」とか「ふりまね」を意味したが、最近では「模擬実験」や「訓練」の意味で使われるのが一般的である。実際に現象を起こさずにあるいは現象が起きる前に、シミュレーションによってその経過や結果を予測することで現象を事前に理解することができ、必要な対策を講じることも可能になる。

コンピュータ・シミュレーションとはコンピュータを使ったシミュレーションのことであり、特に工学の諸分野ではコンピュータの発展・普及とともに広く利用されてきた。例えば、自動車の衝突実験に代表される CAE や気象予報、大規模集積回路の設計、配電・給電計画、経営戦略などコンピュータ・シミュレーションが使われる分野は枚挙にいとまがない。

さまざまな現象をコンピュータの中で再現するためには数学モデルが不可欠となる。数学モデルによってシミュレーションの妥当性や精度が決まるという意味で、数学モデルの構築はコンピュータ・シミュレーションにとって最も基本となる重要な技術である。

ここでは科学技術計算分野の代表である数値流体力学の例を通してシミュレーションとモデリングについて概説する。これまで主であった機能指向モデリングと、これとは異なる概念に基づくオブジェクト指向モデリングの違いを示し、科学技術計算における後者の意義と現状について紹介する。

### 2. 数値流体力学

水や空気に代表される流体の運動は一般に場の概念で表現される。これは、対象を粒子の集まりとして記述する粒子的描像とは異なり、圧力や流速などの物理量が空間内に分布するという考え方である。流体力学に限らず電磁気学や弾性力学でも場の概念は使われており、これに対する数学モデルは一般に偏微分方程式となる。流体の運動法則に対する数学モデルは Navier-Stokes 方程式と呼ばれる非線形偏微分方程式系である。

Navier-Stokes 方程式自体は 1827 年に C. H. Navier が、続いて 1845 年には G. G. Stokes がそれぞれ独立に導いた。この方程式を適当な初期条件と境界条件の下で解けば原理的には流体の運動が決まるが、その具体的な解は特殊な場合を除いてほとんど知られていない。したがって、この方程式の一般的な解を得る唯一の方法はその数値解を求めることであり、そこから数値流体力学と呼ばれる分野が発展した。

数値流体力学に限らず偏微分方程式の数値計算においては、時間発展の有無のかかわらず反復計算が用いられる。そして、その基本的な処理の手順自体はきわめて単純である。それは、まず物理量に対する初期値の設定を行い、その後に境界条件の設定と基礎方程式による物理量の更新という手順を繰り返すことで構成される。このように科学技術計算で用いる計算手順はそれ自体の制御構造が比較的単純で構造化やパターン化に適している面がある。そのため、従来のモデリング手法である機能指向方法論の適用が容易であった分、この方法論がもつ原理的な制限に関してほとんど意識されてこなかった。

扱う系のもつ機能に着目し、分析・設計する方法論を機能指向方法論といい、コンピュータ・シミュレーションが発展した 1970 年代から 80 年代にかけてモデリングやプログラミングで主流の考え方であった。この方法では、トップダウンの機能分割、すなわちソフトウェアが持つべき機能の細分化を繰り返すことによりソフトウェア全体を設計する。したがって、プログラムの各部分は呼び出す側と呼び出される側という主従関係によって結ばれ、その処理形態は逐次的、直列的となる (フローチャートによる表現)。

機能指向方法論がもつこのような処理形態が従来のハードウェアの反映であることは明らかである。つまり、ソフトウェアあるいはモデリングにおいて、その基本となる考え方がハードウェアによって制限される。特に機能指向方法論がもつ逐次性や直列性は古典物理の時間概念と直接に結びつくため、それが上述した偏微分方程式のアルゴリズムの基本となった。そして、この考え方は時間とは対照的に一定の方向性をもたない空間領域の処理にも適用される。つまり、空間的に分布する物理量の決定にも通常は反復処理が使われる (否しか使えない)。しかし、空間方向の反復処理は情報の流れを *a priori* に与えることになり、対応する物理現象の忠実なモデルになるとは限らないことに注意すべきである。

さらに、演算速度向上にともなって並列計算が普及するようになると、この点が大きな制限となった。機能指向方法論の基本は逐次処理であるため、計算を並列化するには反復処理自体を分割する以外に方法がない。そのため、場全体を小領域に分割して各領域で場の量を計算する方法 (SIMD) が並列化のための唯一の指導原理となる。

### 3. オブジェクト指向モデリング

オブジェクト指向方法論では、系を構成する対象(object)とそれらの相互作用として系を記述し、それに基づいて分析・設計を行う。わが国でも 1980 年代後半から 90 年代前半にかけてオブジェクト指向方法論が普及した。これを受けて機能指向技術が主流であった科学技術計算の分野でも 1990 年代後半からオブジェクト指向方法論を応用した研究結果が報告されるようになった。科学技術計算の中でも特に必要性の高い偏微分方程式の数値計算に対するオブジェクト指向技術の応用は 2 つの方向に大別できる。

ひとつはオブジェクト指向方法論に基づくクラスライブラリを提供することで計算環境としての汎用性や連携性、並列化の容易さを向上させることを意図している。ここでは実用的側面が強調され計算機環境とのインターフェイスに重点が置かれるため、オブジェクトそのものの自律性や相互作用といった点は副次的な扱いになる。つまり、従来の機能指向モデルを効率の良い並列計算環境に持ち込むための便法としての意味合いが強い。

一方、他の方向性としてモデリングそのものに重点を置いた応用も可能である。この場合、系を構成する最小単位をオブジェクト化することで自律的オブジェクトの相互作用として系が表現される。その意味で、この方向はオブジェクト指向方法論の忠実な実践といえる。このモデルでは、モデル自体が自律性をもつのでそれらを分散化すればそのまま並列計算にもなる。したがって、逐次/並列といったハードウェア環境とは独立にソフトウェアを開発できる可能性がある。

後者の観点に立ったオブジェクト指向モデルとして、(1)サッカー競技、(2)市街地での交通流、(3)領域オブジェクトを取り上げ、それらのシミュレーション例を紹介する。そして、それぞれのモデルの特徴を明らかにするとともに、科学技術計算への応用を考える際の問題点や課題についても言及する。