

## 解説



## コンピュータグラフィックス応用の発展動向†

國井 利 泰†

## 1. はじめに

コンピュータグラフィックス (CG) 技術の研究は、一般には 1963 年 MIT のサザーランドによって開発された Sketchpad<sup>1)</sup> に端を発しているといわれている。しかしこの Sketchpad 自身は、CG というよりも現在でいうところの CAD (Computer-Aided Design) を目的としたシステムであった。つまり、CG 技術の誕生は、そもそもがその応用分野からの要請によってもたらされたものであったといえる。しかし CG 本来の姿としては、CAD のみに限らずもっと豊富な応用分野を花咲かせる可能性を秘めたものであった。

CG 自身とその応用分野との密接な関係は、その誕生の経緯のみに留まらず、その後の発展においても容易に見出すことができる。CG のソフトウェアだけでなくハードウェアについても、若干の程度の違いがあるものの、同様な傾向を見て取ることができる。線画のもつ表現力に飽き足らないユーザからの要請によってラスターグラフィックスの技術が生まれ、そのためのソフトウェアの研究が盛んに進められた。これに並行して、半導体技術など、これらの研究成果を低価格で実現するために必要となるハードウェア技術も進歩を遂げてきた。

このような経緯を経て、CG は飛躍的な進歩を遂げた。それと同時にその応用分野も拡大し、本特集の内容からも明らかのように、今日では CAD のみに留まらず実に多岐にわたっている。ここでは取り上げられていない分野も数々ある。研究・開発の対象としてはあまり関心を集めないながらも市場としては最大規模を占めるものとして、事務書類用の図面とかグラフ作成のためのオフィス・グラフィックスがある。現在ではパソコン用のソフトウェア・パッケージとして供給されている場合も多いようだが、これさえも一昔前

は夢物語だった部分が多い。このようなことが現実となったのも、ハードウェアならびにソフトウェアの両面における CG 技術の発展によるものである。

CG 技術の応用はさらにその間口を広げつつある。その反面、現在の状況に対して批判的な見解も出されている。CG の応用分野として最も古い歴史をもち、最も成熟した分野と考えられている CAD できさえも、当初その目指した目標を完全に達成しているとはいえない。なかには、CAD を称して「電気烏口」と酷評する向きさえある。

そこで本稿では、CG の応用分野で用いられてきた従来のアプローチを再検討し、それにとりなう問題点を洗い出すことから出発する(2.)。3. では、その問題点を解決する一つの方法として筆者らが提案している、モデルに基づく新たなアプローチについて述べ、4. においてその簡単な適用例を紹介する。最後に 5. では、情報処理の他の分野(データベース・システム)とのアナログについて考察し、今後の展望を述べる。

## 2. CG 応用への従来のアプローチ

ここでは、機械系 CAD、すなわち機械設計作業への CG の応用を例に取って、そこで取られているアプローチを検討する。CAD を例とする理由は、先にも述べたとおり、この分野は CG の応用分野としては最も成熟した分野と考えられるからである。

CG とは可視化の技術であり、可視化の効用が大きいところで活用される。機械設計作業において、文字以外の情報を多く含み、可視化の効用が大きいと考えられる対象はなんだろうか。第一候補にあげられるのは、設計図などの各種「図面」である。

CG 技術が導入される以前は、作図作業はすべて人手によって行われていた。ここでは、製図板、定規、烏口などの「ツール」が使用されていた。こうして描かれた図面は、一度作成されてしまえばその時点で硬直化してしまう。変更が生じたり図面自身が破損したりした場合には、その大小にかかわらず、最初から新

† Advances in Computer Graphics Applications by Tosiyaasu L. KUNII (Department of Information Science, Faculty of Science, The University of Tokyo).

† 東京大学理学部情報科学科

たに書き直さねばならない。つまり、図面の再現性はすべて人手に委ねられていたといえる。

その後CG技術が導入されたことにより、上記のツールは、グラフィックス端末のスクリーン、タブレット、スタイラスペンなどに取って代わられた。一度作成された図面はデータベースに登録され、必要に応じて取り出すことが可能となった。この結果、図面の再現性がコンピュータによって保証されるようになった。同一の図面を作成しただけでは、データベースからその図面を読み出してきて、それをプロッタなどの出力装置を用いて出力するだけで済むようになった。変更も容易に行えるようになり、作図作業にともなう時間とコストが大幅に削減された。

ここでのCGの役割には、プログラム開発におけるオンライン・エディタのそれに似たところがある。オンライン・エディタ以前は、カードを媒体としてプログラムを記述していた。カードへの穿孔のツールとしてキーパンチが与えられていたが、これは省力化を目的としたものであり、本質的な意味でプログラミング環境を変えるものではなかった。カードが破損したり、その内容に変更が生じたりしたときは、また新たなカードを作成しなければならなかった。

これらの問題点は、オンライン・エディタによるプログラム作成環境の大幅な改善により解決された。作成されたプログラムはコンピュータの二次記憶に蓄えられ、更新も容易に行えるようになった。この結果、(少なくとも理論的には)ハードコピーを一切用いることなくプログラム開発を行うことが可能となった。

設計作業に話を戻そう。設計作業とは、単に作図を行うだけのものではない。解析や試験といった作業を並行して行い、その結果を新たな設計案へ反映させなければならない。したがって、設計作業によって作成された図面は、これらの作業に携わる人間によって再び解釈された上でなければその意義をもたない。つまり、設計対象は設計者の解釈に基づいて図面の形にまとめられるが、この図面は、他の作業に従事する者によって再び解釈されなければならない。同一の対象をめぐってこのように再度解釈をほどこす必要性という点に関しては、CGを導入した後も、それ以前と一向に変わっていない。

オンライン・エディタと一見似た役割をもつCGでありながら、その効用という点についてみてみると、大きな違いがあることが分かる。オンライン・エディタは、統合的なプログラム作成環境を築くために大い

に役立っている。ところが、統合的な作業環境という観点から設計作業全般を捉えた場合、CG技術も現状ではまだ十分にその機能を果たしているとはいえない。この違いはどこに起因しているのだろうか。

オンライン・エディタを用いたプログラム作成環境では、エディタをとおして操作する対象(原始コード)はそのままの形で次の工程(コンピュータによる解釈)に回され、処理される。ユーザがエディタを通じて操作する対象と、コンピュータが解釈する対象とは同一のものである。しかも、すべての工程を1台のコンピュータの内部のみで処理している。いい換えれば、ここにはすべての工程に共通したモデルが存在し、各工程ではそのモデルをその工程なりの見方に基づいて操作している。

それに対して、設計作業におけるCG技術は基本的には単なる「表示」のみを目的としており、関連したすべての工程に共通したモデルを直接操作するものとはなっていない。

以上の議論をまとめると次のような結論が導かれる。つまり、CG技術とエディタ技術の効用の差異は、その操作対象となるモデルのレベルに起因し、より高いレベルのモデルを扱うエディタ技術のほうが、CG技術に比べると大きな効果をもたらす結果となっている。

CAD以外の応用分野でも、CGの応用は表示のみを意識した形で進められてきている。いうなれば、CG応用への従来のアプローチは、「表示指向的(presentation-oriented)」であったといえる。

### 3. モデル駆動型アプローチ

前章では、従来のアプローチの問題点として、表示指向的な性質を指摘した。では、このような問題点を回避し、同時にCGのもつ可能性を最大限に引き出すアプローチとはどのようなものであろうか。われわれはその一つとして、「モデル駆動型(model-driven)」アプローチ<sup>2)</sup>を提案してきた。モデル駆動型アプローチがどのようなものであるかを論じる前に、まずモデル階層について若干の考察を加えておく必要がある。

#### 3.1 モデル階層

今までみてきたとおり、CGはそれ単独では視覚情報処理における表示のみを受けつつものに過ぎず、統合的な環境で用いることによって初めて、その全可能性を引き出すことができるものである。

統合化の対象となる各要素技術は、それ独自のモデ

ルに基づいて発展してきたものである。そこで統合化においては、これらの複数のモデルを同一の枠組みに基づいて扱うことになる。

関連し合った複数のモデルが同時に存在する場合、その間に階層関係の存在が認められることが少なくない。この階層関係は、モデル間の親子関係を、次のような規則の適用を繰り返して確定することにより作り上げることができる<sup>2)</sup>。

(1) 抽象化規則：抽象度の異なるモデル間の汎化階層。抽象度の高いモデルが親、抽象度の低いモデルが子となる。

【例】 画像意味モデル ≧ 幾何学モデル  
    ≧ ピクセルモデル

(2) 包含規則：構成要素モデルの集約化による階層。構成要素となるモデルが子、集約された結果のモデルが親となる。

【例】 図面モデル  
    ← 線画モデル + テキストモデル

### 3.2 モデル駆動型アプローチとは

要素技術をまとめ上げ統合的な環境を構築する際には、上記のようなモデル階層を扱うことになる。統合環境の中核となるモデル（マスタ・モデル）自身がこの階層構造の最上位に位置するような形で統合化を図るアプローチを、モデル駆動型アプローチと呼ぶ。

このアプローチによって構成された環境では、中心にマスタ・モデルが据えられ、各要素技術がそれを取り巻くように配置される。ここでのマスタ・モデルの役割は、共通データを集中管理し、各要素技術に対してそれに適した「視点（ビュー）」を提供することである。この視点は双方向性を持ち、それを通じてマスタ・モデルから情報を抽出できるだけでなく、新たなモデルの生成ならびに既存のモデルに対しての更新も行えるようになっていなければならない。

表示指向的なアプローチと比較した場合、モデル駆動型アプローチのもつ利点として次の2点をあげることができる。

#### (1) 単一の共通モデル

表示指向的なアプローチでは、異なる工程間で受渡されるのは表示の結果である。その表示結果は、それを生成した工程の性質を反映したものであり、受け取った側にとっては必ずしも扱いやすいものとはなっていない。そのため、再解釈の必要が生じる。これは工程全体の進行を遅延させるだけでなく、解釈の違いによる誤解が介入する可能性を生じさせてしまう。

すべての作業工程において共通のモデルを使用するモデル駆動型アプローチでは、工程間で受渡しされる情報は存在しない。そこに存在するのは、すべての工程に共通した単一のモデルである。各工程では、作業の進行状況に応じて順繰りにこのモデルをアクセスし、必要となる情報を抽出したり、変更を施したりする。

#### (2) データ入力誘導

表示指向的なアプローチでは、2次元のスクリーン上に図形を描くことに主眼がおかれている。その図形が何を意味するかはユーザのみの知るところである。必要となるデータはすべて、ユーザ自身が入力しなければならない。そのため、ユーザの不注意によって、まったく意味のない図形を描いてしまうことも十分に起こりうる。扱う図形が複雑になればその可能性も増大する。

それに対してモデル駆動型アプローチでは、マスタ・モデル自身の構造が、必要となるデータの枠組みを規定している。モデルのもつ整合性などの制約によって、一部のデータだけが入力された段階で、残りのデータの取りうる可能性が少数に限定されてしまう。したがって、モデル自身からユーザに対して特定のデータ項目の入力を要求することにより、モデル構築のための手順を大幅に短縮することが可能となる。それに加え、新たに入力されるデータをモデルのもつ制約条件と照らし合わせることによって、不正データの検出を行うことができるので、できあがったモデルの信頼性も向上する。

中核となるモデルを具備することによって、不必要な情報交換を廃絶し、ユーザに対する不要な負担を軽減するのが、モデル駆動型アプローチの最大の利点である。このような環境において初めて、CG技術のもつ力も最大限に発揮されるようになる。

このようなアプローチ自身は、CG技術の範疇のみ含まれるものではない。しかし、CG技術のある分野に適用し、その力を十分に引き出すためには必要不可欠なものである。CG応用においては必須の、統合化のための技術と呼ぶことができよう。

### 3.3 モデル駆動型アプローチの手順

CG技術の応用を意図してモデル駆動型アプローチを適用する際には、以下のような手順を踏む必要がある。

#### ① 統合化の対象となる範囲の設定

CG技術を応用することによってどのような要素技

術を統合するかを決定する

② 情報要求ならびに処理要求の収集と分析

③ 仕様決定と分析

必要となる CG 技法はこの時点で決定される

④ マスタ・モデルの構築

マスタ・モデルは、個々の要素技術がもつ独自のモデルの最上位に位置している必要がある

⑤ モデルによるデータ入力誘導を用いた効果的な入力系の決定

実際のシステムを構築するためにはもっと多くの手順が必要となるが、このアプローチの特徴となる点は上記の順に含まれている。一見するとごく普通のシステム設計手順のようにみえるが、各要素技術に独自に備わっているモデルからマスタ・モデルを構築するという点に違いがある。

マスタ・モデルは、以下の三つの要素から構成される。

(1) モデルのもつ情報構造

(2) 上記の情報構造に対して施される操作群

(3) モデルに課せられた整合性規則

これら三つの要素は、要求より導出された仕様を満足するものでなければならない。(2)の操作群としては、処理要求を満たす最小の組合せで十分であるが、使い勝手を考えて、その他のものをいくつか付加するのが一般的である。

#### 4. モデル駆動型アプローチの適用例

##### 4.1 マスタ・モデル構築のための二つのアプローチ

統合化環境のためのマスタ・モデルの構築に関しては、2通りの方法が考えられる。一つは、各要素技術で用いられているモデルから出発し、それらを融合してボトムアップに一つのモデルにまとめ上げるものである。この方法だと、統合後もおのおの要素技術を変更なしに活用することができる。しかし、おのおのに備わっているモデルは元来統合化を意図したものとはなっていないので、融合の作業において多くの困難をとらう。

逆にトップダウンに考える場合には、新たなマスタ・モデルを構築し、そのモデルと各要素技術との対応関係を取ることによって統合化の作業が進められる。この場合、各要素技術で用いられていたモデルがそのまま有効であるとは限らない。全範囲に及ぶ新たな技術体系を、マスタ・モデルに基づいて築き上げる

必要がある。しかし、その際の基本思想がマスタ・モデルの形で具体的に示されているので、一貫性の取れた環境を作りやすいという利点がある。

##### 4.2 機械設計への適用例

現在までのところ、ここで述べたことを完全に満足させることは難しいが、機械設計の分野などでは、その長い歴史にも支えられてかなりの成果が得られている。そこで、まず機械設計を例にとってモデル駆動型アプローチを実際に適用した場合を考察する。

機械設計においての主な処理対象はその形状である。ネジ、歯車といった機械要素の場合だと、形状によってその機能、性能のほとんどが決定されてしまう。機械設計においての設計対象は3次元の剛体であることからソリッド・モデル<sup>3)</sup>がマスタ・モデルとしての役割を果たしうることが知られている。ソリッド・モデルを用いたアプローチはトップダウンなアプローチであり、各要素技術はこのモデルに基づいて再編成される。

ソリッド・モデルを用いた場合の、モデル駆動型アプローチによる CAD 環境を考えてみよう。ここでは、ソリッド・モデルに基づくデータベースが全工程に共通のものとして用意され、新たなモデルの生成、ならびに既存のモデルへの変更の操作などはすべてこのデータベースに対して行われる。この際、処理結果をグラフィックス表示することによって、ユーザは自分の行った操作の確認を容易に行うことができる。次の工程に進むと、別のユーザが同一のデータベースをアクセスする。ユーザごとに特定の視点(ビュー)が設定されているので、ユーザの要求する情報を望みどおりの形式で提供することができる。

設計作業の難しさの一つに、設計対象は実際には存在しないものであるということがある。この対象はデータの形でデータベース内に蓄えられてはいるが、その実態はどこにも存在しない。このような実態のないものをあたかも実存するものであるかのように見せるのが、CG の役割である。ユーザに対象物を直接操作しているかのように思わせるのは、CG 技術なしではまず不可能なことである。

そのほかにも、各種解析結果の数値データをグラフなどの形式に図式化するのも、CG の役目の一つである。また、CG を用いたシミュレーションも、単なる数値シミュレーションよりもはるかに効果的な結果をもたらしてくれる。このような CG の機能も、統合化環境で用いられた場合には、単一で用いられたときよ

りもずっと大きな威力を発揮する。

これまでの議論から、レイ・トレーシングなどのレンダリング技法はあまり重視されていないのではないかという印象をもたれる読者もいるかもしれない。決してそのようなことはない。実存しないものを目の前にあるかのように見せることに関しては、レンダリングは必要不可欠な技術である。特に意匠設計などの段階では、ほかのどのCG技術よりもレンダリング技術が重視される。この際も、それまでの設計作業に用いたマスタ・モデルをそのまま用いてレンダリングを行うことができれば、その効果はずっと大きくなる。

機械設計の場合は単に形状を扱うだけで十分であった。透明もしくは半透明な物体の場合は、特に宝石の場合は、光の透過や分散とカッティングとよばれる形状との間に密接な関係がある。このような場合、上で述べたレンダリング、特に色分散を考慮した光線追跡法と組み合わせることが重要になる<sup>9)</sup>。半透明なものの場合は、内部での色散乱も考慮しなければならない。

#### 4.3 地理情報システムへの適用例

もちろん、このモデル駆動型アプローチは、3次元の情報ばかりでなく2次元の情報の場合も重要なシステム設計上の指針となる。たとえば、地理情報システム(GIS)の場合である。GISでは、行政区分の境界や道路などのベクタ型のデータと、領域を示すラスタ型のデータの2種類を扱う必要がある。これら2種類のデータはその操作群にも大きな違いがあるが、両者を同時に扱う操作も考えられる(例:ある領域内にある道路を求める)。このような場合には、マスタ・モデルとして両者の特性を同時に備えているものを用いるのが好ましい。こうすれば、ベクタ型データに関する操作、ラスタ型データに関する操作のほかに、両者を同時に扱う操作も容易に実現することができる<sup>9)</sup>。

#### 4.4 医療画像処理への適用例

扱われるモデルがさまざまであり、ボトムアップにマスタ・モデルを構築しなければならない場合として、医療画像処理がある。まず取り扱わなければならないモデルはビットイメージであり、これにある種のコヒーレンスを考慮したクアド・ツリーなども一種の中間モデルであるといえる。その段階から注目すべき領域を抽出することは、事実上上位のモデルへの変換といえるが、連結成分発見アルゴリズムを用いた雑音消去法が提案されている<sup>6)</sup>。また、モデルに応じてレンダリングの方法も考慮されなければならない。クアド・ツリーを3次元的に再構成したオクト・ツリー、正確

には多値のオクト・ツリーにおいて半透明表示を行うことも可能である<sup>7)</sup>、また、境界がパッチで表現されている場合は別の陰面処理技法が必要である<sup>8)</sup>。

#### 4.5 スポーツ・インタラクションへの適用例

さらに、このようなアプローチの適用範囲は従来のCGの適用範囲に留まらない。スポーツ・インタラクションを目的としたシステムでは、スポーツを行う主体である人間自身、スポーツ用具、および環境のモデル化を行う必要がある。スキーを例にとると、関節とセグメントから構成される人体モデル<sup>9)</sup>のほか、スキーウェア、ストック、スキー、さらに雪面のモデルの構築が必須である<sup>10)</sup>。

### 5. 今後の展望

モデル駆動型アプローチを用いる際最も重要なことは、適切なマスタ・モデルの決定である。先に述べたように、機械形状の設計などにおいてはソリッド・モデルがマスタ・モデルとなりうるものとして知られている。しかし、ソリッド・モデルだけを取ってみても、CSG(Constructive Solid Geometry)法と境界表現(B-rep)法が知られており、ともに長所短所を合わせもっていて、一つには限定しきれないのが現状である。また、形状設計のみに限定せず機能設計との融合を図ろうとすると、いずれも不十分であるといわれている。

そのほかの応用分野に関して考えてみると、いまだにソリッド・モデルに当たるようなマスタ・モデル候補は見つかっていない。一言でいうと、単に多数の応用が個別に開発されたという状況にある。今後は、各分野におけるマスタ・モデルに関する研究が期待される。

ここで視点を変えて、情報処理の他の分野に目を向けてみよう。「モデル」の概念が積極的に使われている分野は数多くあるが、そのなかでもデータベース・システムの分野では、理論的な基盤の上に成り立ったモデルの概念を扱い、大きな進歩を遂げてきた。よく知られているように、データモデルの概念に先鞭をつけたのは、1970年コードによって成された関係型データモデルの提案である<sup>11)</sup>。その後、数々の困難はあったものの、この分野は飛躍的な進歩を遂げた。その結果、1980年代初頭あたりから、関係型モデルに基づく実用的なシステムも現れはじめた。

このような大幅な進歩を遂げられたことは、関係型モデルのもつ論理的な基盤に負うところが大きい。し

かし、当時は究極のデータモデルであるかのように思われていた関係型モデルも、最近ではその限界が指摘されるようになり、さらにそれを越えるデータモデルの研究が行われている。

データベース発展の経緯と照らし合わせてみると、CG 応用の現状はデータベースの1970年以前に相当するといえるかもしれない。モデリングの重要性が認識され、皆がそれに気づき始めた頃である。今後は実際にどのようなモデルが有効かを考える時期といえるだろう。ただし、ここで扱う問題は複雑多岐にわたり、関係型モデルのような一つのモデルに限定して研究を進めていくことはおそらく不可能であろう。また、ふさわしいマスタ・モデルが見つかったとしても、さらに応用分野が広がるにつれて、それだけでは不十分になってしまう可能性もある。解決されるべき問題は、いまだ山積みされたままといったところだろうか。

## 6. ま と め

これまでのCG 応用の経緯を振り返り、今後へ向けてのアプローチとしてモデル駆動型アプローチを提案した。モデルを中心に据えたアプローチというのは当り前のことのようにも思える。しかし、これまでのCG 技術の応用においては、文字どおり「眼の前の絵」に注意を奪われてしまった結果、それさえもおろそかにしていた傾向がある。

最近では、プログラミング一般と同様に、CG の分野でも「オブジェクト指向」などといった新しいパラダイムの導入が研究されつつある<sup>12)</sup>。これからは、情報処理分野全般に眼を向け、「眼の前の絵」だけにとらわれない広い視野をもつことが必要となる。

## 参 考 文 献

- 1) Sutherland, I. E.: SKETCHPAD: A Man-Machine Graphical Communication System, SJCC, pp. 329-345 (1963).
- 2) Kunii, T. L.: A Model-Driven Approach to CAD and Graphic Communication Networks, in Techniques for Computer Graphics, D. F. Rogers and R. A. Earnshaw (eds.), Springer-Verlag, pp. 67-107 (1987).
- 3) Requicha, A. A. G.: Representations for Rigid Solids: Theory, Methods, and Systems, Comput. Surv., Vol. 12, No. 4, pp. 437-464 (1980).
- 4) Yuan, Y., Kunii, T. L., Inamoto, N. and Sun, L.: GemstoneFire: Adaptively Dispersive Ray Tracing of Polyhedrons, Tech. Rep., 88-003, 東京大学理学部情報科学科 (1988).
- 5) 曹, 立元, 國井: GISのためのラスタ/ベクター複合データモデル, 情報処理学会第36回全国大会論文集, 2F-4 (1988).
- 6) 前田, 藤代, 茅, 國井, 渡辺: 連結成分発見アルゴリズムを用いた階層的多値画像の雑音除去法, 情報処理学会第36回全国大会論文集, 3W-3 (1988).
- 7) 茅, 藤代, 袁, 國井: 多値オクト・ツリー画像の半透明表示法, 情報処理学会第36回全国大会論文集, 4W-4 (1988).
- 8) 品川, 國井, 野村, 奥野: CT連続断面画像からの立体の再構成用の陰面処理, 情報処理学会第36回全国大会論文集, 1Z-4 (1988).
- 9) 李, 白井, 國井: 動作の定義による三次元の人体アニメーション, 情報処理学会第36回全国大会論文集, 7Z-1 (1988).
- 10) 國井: ビジュアル・コンピュータによる実用スポーツ・インストラクション・エキスパート・システムのアーキテクチャ, 情報処理学会第36回全国大会論文集, 7Z-6 (1988).
- 11) Codd, E. F.: A Relational Model of Data for Large Shared Data Banks, Comm. ACM, Vol. 13, No. 6, pp. 377-387 (1970).
- 12) 藤阪, 河合, 大岩: オブジェクト指向パラダイムによる画像生成, WOOC '88, 日本ソフトウェア科学会 (1988).

(昭和63年6月9日受付)