

教育用 3 次元ソリッドモデラーの開発

Development of 3D-Solid Modeler for Educations

新 津 靖
Yasushi NIITSU

東京電機大学情報環境学部 (〒270-1382 千葉県印西市武西学園台 2-1200
E-mail: yniitsu@sie.dendai.ac.jp)

ABSTRACT. Authors have been developing the 3D solid modeler (Solid Interpreter) for the education of descriptive geometry and computer graphics in this 10 years. The developed software adopts the boundary representation and the winged-edges method as a description method of the solid models. The developed solid modeler can create a three-dimensional solid model by executing the special command scripts of text format. Combining the 37 kinds of primitive models, the optional model is constructed for examples the house, automobile, ship, human, animal and so on. The boolean operations, affine transfer operation are also variable. The developed software has several kinds of display modes more than 140, such as the edge-type, the surface-type, the perspective and parallel views, three faces and so on. The practice of descriptive geometry with the developed software had been carried out in Tokyo Denki University in 1999, 2000 and 2001. The students were interested in the geometry, and their geometric ability was improved. Recently, many Japanese Universities adopted our software for the education of the geometry and computer graphics. Authors want to apply this software to the mathematics education of elementary school and junior high school. Recently, authors have developed the function of the automatic expansion on the sheet from three-dimensional model. The students can make the real models of the paper craft.

1. 背景と目的

著者が所属する教育機関では、10年前、1年次生のコンピュータ教育の充実を機に、1年次の図学授業を廃止し、コンピュータ授業の中で図学教育を行う方向でコンピュータ教育のカリキュラムを組み、2次元および3次元グラフィックスのプログラムおよびデータ作成を行った。3次元グラフィックスでは投影法、ワイヤーフレームモデル、サーフェイスモデル、陰線処理、相貫線表示を行っていた。しかし、教育システムがMS-DOSからWindowsに変わるとのに伴い、ワープロや表計算、数式処理などのリテラシー教育に多くの時間が割かれるようになり、プログラミング教育に使える時間が減少してきた。その結果、プログラミングによる図学教育を実施することが困難になってきた。このような現状から、図学教育を意識したコンピュータ教育を一年次に行うには、初心者でも操作できる図学教育用ソフトウェアを開発および導入する必要があった。

そこで著者は、使いやすく教育的にも配慮された図学教育やコンピュータグラフィックス教育などに応用できる3次元ソリッドモデラーを開発した。開発したソフトウェアは、図学教育に援用することを考慮し、3次元CADに近い機能を付加した。すなわち、集合演算と陰線処理を充実させた。そして、1999年と2000年にこのソフトウェアを使用したコンピュータ図学教育を行った。その結果、十分な教育効果が認められた。現在、その授業を

受けた学生たちは上級学年で設計製図などの授業を終えているが、その担当教員からも教えやすいという意見をj得ている。また、2001年度は、情報環境学部の1年次生の入学直後のワークショップ授業(学生が教員と一緒に個々のテーマを設定してそれを遂行する実験実習授業)でソフトウェアを使用した。

上記のバックグラウンドとして開発したソフトウェア「Solid Interpreter」は、平成12年度の「未踏ソフトウェア創造事業」に採択され、GUIの改良と大幅な機能の向上を図ってきた。そして、平成13年度には、ソリッドモデラーの機能向上に加えて、展開図の生成機能と外部入出力機能を追加した。

2. 三次元ソリッドモデラー

開発した教育用3次元CAD「Solid Interpreter」は、テキスト形式の独自のスクリプト言語を解釈してモデルの構築、演算、表示を行うソフトウェアである。このソフトは、基本立体と呼ばれる13種類の立体に移動や回転、アフィン変換、集合演算^[3]などを施して希望とする3次元立体を構築することができる。ソフトウェアの開発で留意した点は、初心者でも楽しく使えることと、図学教育が目的とする「空間把握能力」を高める効果が得られることである。

2.1 ソリッドモデルのデータ構造

コンピュータ内部のデータとして3次元立体を表現する方法には2種類の方法がある。一つは立体の境界条件である面や稜線、頂点を用いて立体を表す B-Reps 法 (Boundary Representation) であり、もう一つは円柱や直方体など直感的にわかりやすい基本的な立体の組み合わせで目的の立体を表現する CSG 法 (Constructive Solid Geometry) である。CSG 表現は直感的に理解しやすい表現方法であるが、集合演算や投影表示の容易さや、市販のほとんどの3次元 CAD と有限要素解析用プリポストプロセッサが B-Reps 法を採用していることを考慮し、本ソフトウェアも内部形状表現に B-Reps 法を採用した。^{[1],[2],[3]}

開発したソフトウェアは3次元立体の生成、表示、管理を行うもので、ソフトウェアが扱うデータには、名前を付けて管理するものと、名前を付けずにポイント操作で管理するものの2種類がある。名前を付けて管理するデータ構造としては、表1に示す4種類のものがある。

視線情報は、立体を表示するときの条件を設定するためのものである。表示モードには、面表示^[2]と稜線表示モードがあり、陰線の表示・非表示を指定することができる。^[6] また、3面図表示により通常の2次元設計図面に対応した表示を行うことができるように設計した。このため、非常に多様な表示モードが指定可能であり、その数は140種類以上にもなる。表示モードは4つのパラメータで表現される。グループ構造は立体や子グループをまとめて扱うためのものであり、コンピュータのファイル構造に似せて設計した。たとえば、2つのグループを作り、その中に同じ名前の立体を作ることができる。あるグループを表示コマンドで表示する場合、グループ内部のすべての立体とすべての子グループさらに孫、ひ孫グループが表示される。軸構造体は、立体やグループに動作を定義するためのものである。たとえば、飛行機のプロペラを考えて見ると、プロペラは飛行機に接続して回転するが、このような動作を軸構造により表現することができる。

立体は基本的に多面体として表現され、そのデータ構造には、Winged-edge データ構造を採用している。^{[3],[4],[5]} すなわち立体の境界を表す面の構造体と稜線の構造体のリスト構造により立体の位相関係を表現する方法である。立体を構成する要素である「頂点」、「稜線」、「面」、「ループ」は名前を持たないデータ構造である。高速処理を実現するため、位相データの検索・操作はポイント操作により実現している。

2.2 3次元立体の形状表現

立体は内部にその立体を構成する面構造体のリストを情報としてもっており、面を順次表示することで、立体を表示させることができる。面、稜線、頂点、Loop は立体を表現する位相データであるが、これらは名前前で管理せず、ポイントのリスト構造により管理されている。

コンピュータ内での立体の表現方法は、構造体のリスト構造を用いている。Body (立体) はすべての Face (面) へのポイントを持っており、各 Face は最初(外側)の Loop へのポイントとすべての Loop を結ぶ一方向リストを持っている。Loop は最初の Edge へのポイントを持っており、Edge (稜線) は左右の Face と Loop へのポイントお

	ード, 表示効果
グループ	グループ名, 子グループリスト, 子立体リスト, 内包軸ポイント
軸	軸名, 関連立体/グループ名, 軸方向ベクトル, 移動量, 上下リミット
立体	立体名, 面色, 稜線色, 親グループ, 境界面・稜線・頂点 Class 等へのポイント, 面数・稜線数・頂点数等

よび始点・終点の頂点へのポイントを持っている。^{[2],[4],[5]} これにより、1つの Edge を2つの Face と Loop で共有することができる構造になっている。このように、Face が複数の Loop (親 Loop と複数の子 Loop) を持つことができ、この位相構造により穴の空いた面を表現することができる。

2.3 プリミティブ(基本)立体の種類と形状

B-Reps 法による立体表現は、コンピュータにとっては処理しやすい方法であるが、人間が直接データを入力するのは不可能である。このため、図1に示すように、6種類(直線, 多角柱, 多角錐, 2種類の回転体, 螺旋体)の立体生成コードを用いて生成できる13種類(直線, 直方体, 円柱, 円錐, 球, 回転楕円体, トーラス, 多角柱, 多角錐, 2種類の回転体, 2種類の螺旋体)の基本立体を考え、12の立体生成コマンドとパラメータ指定により簡単に基本立体が生成できる方法を取った。基本立体のうち、直線は面を持たない特殊な立体である。図1の矢印は、上の基本立体と同じ生成プロセスで下の基本立体を生成していることを示している。実際にはこれに加えて正12面体, 正20面体, 32面体, 80面体など、合計で37種類の基本立体を図1の13種類のコマンドとオプション指定で生成することができる。

2.4 表示モードと表示効果

「Solid Interpreter」の特徴はその表示機能にある。CAD や図学教育を意識して開発されたため、陰線処理や3面図表示、リアルな面表示など140種類以上の多彩な表示方法を有している。図2は、トーラスを集合演算により組合せた立体の表示例であるが、陰線処理を施したものと、サーフェイスモデルによる簡易陰線処理の違いが示されている。図3は3面図表示と陰線処理の例である。本ソフトウェアの開発においては、図学や設計の教育での応用を念頭に考えているため、陰線表示と3面図表示は充実させている。表示モードと表示効果は、視線情報

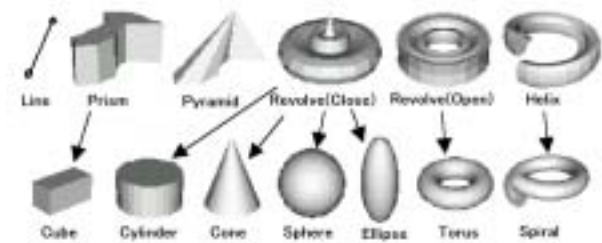


図1 基本立体の種類と形状

表1 名前を持つ Object のデータ構造

Object	Object が有する主な情報
視線情報	視線名, 視点座標, 注視点座標, 表示モ

を生成するとき4つのパラメータで指定する．それぞれのパラメータの分類を下記に示す．

- (1) 表示効果 : 全稜線表示, 簡易陰線処理, 陰線処理, 面表示(3タイプ)
- (2) 表示モード: 透視投影, 平行投影, 3面図(2タイプ), 3面図と透視投影・平行投影の組合せ
- (3) 陰線表示 : 表示 ON・OFF 制御
- (4) 曲面上稜線: 表示 ON・OFF 制御

これらのパラメータの組み合わせにより原理的には192種類のパターンが設定可能であるが, 面表示と陰線表示のように機能しないものもある．

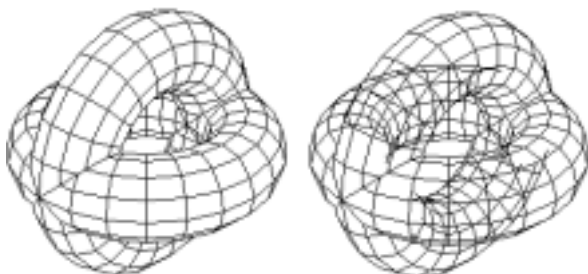
グラフィックス表示はWindowsのAPIを使用して実現しており, OpenGLやDirectXは使用していない．その理由はOpenGL等では陰線処理ができないためである．しかし, 本ソフトウェアの表示エンジンでは, 面表示をZソート法で行っており, 正確な面表示や滑らかな曲面表示ができない．このため, 最新版ではOpenGL等を利用した表示モードを追加している．立体のデータ構造はOpenGLに十分対応できるように設計されており, 技術的には面表示のOpenGL化は比較的容易であった．図4に本ソフトウェアで作成した立体モデルのOpenGL表示例を示す．滑らかな面が表示できていることがわかる．

3. コマンド群と3Dプログラミング

3.1 ソリッドインタープリタのコマンド群

ソリッドモデルの生成と表示はコマンドスクリプトを実行することで実現させることができる．コマンドは47種類あり, 変数や繰返し処理, 条件分岐処理など言語的な機能も備えている．CAD機能として, 立体のグループ処理や立体の動作表現も可能である．表2, 3に, 47種類のコマンドの内, 代表的な20種類のコマンドの機能を示す．コマンド群は大きく分けて以下の7種類に分類できる．

- (1) 立体生成コマンド: 基本立体の生成
- (2) 立体操作コマンド: コピー, 移動, 回転, 集合演算
- (3) 表示機能コマンド: 視線情報生成, 表示制御
- (4) 実行制御コマンド: 繰返し・条件分岐, 変数・定数
- (5) ファイル操作コマンド: ファイル実行, DXF, VRML
- (6) グループ・軸コマンド: グループ化, 動作設定
- (7) その他のコマンド : 初期化, 実行停止, コメント



(a) 完全陰線処理 (b) 簡易陰線処理

図2 完全陰線処理と簡易陰線処理

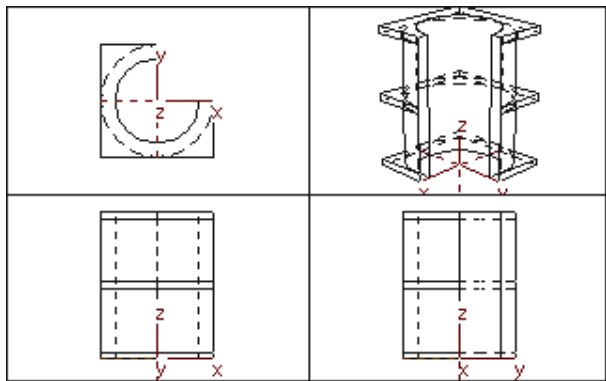


図3 3面図表示と陰線処理

例として, 表2, 3にそれぞれ立体生成コマンドと立体操作コマンドの名前と書式を示す．ほとんどのコマンドはオプション指定によりその機能を変化させることができ, たとえば, 表3のRotateコマンドは立体やグループを回転させるだけでなく, オプション指定により「視点」や「注視点」を回転させることもできるように設計されている．

3.2 3Dプログラミング

簡単な例として, 以下に円柱と円柱の集合演算からできる立体のプログラム例(13行)と実行例(図5)を示す．

```

New # 初期化
Eye eye0 100,100,82, 0,0,0, 3, 1, 4, 0, 0 # 視線生成
Cylinder y0 32, 10, 30 # 円柱 y0 の生成
Move y0 0, 0, -15 # 円柱の移動
y1 = y0 # 円柱のコピー
Rotate y1 0,0,0, 1,0,0, 90 # 円柱 y1 の回転
Xcolor y0 255, 255, 255 # 着色 (白)
Xcolor y1 255, 128, 0 # 着色 (青)
XObj1 = y0 + y1 # 集合演算(和)
XObj2 = y0 - y1; Move XObj2 -25,-25,0 # 集合演算
XObj3 = y0 * y1; Move XObj3 -25, 30,0 # 集合演算
Display eye0 XObj1,XObj2,XObj3 # 表示

```

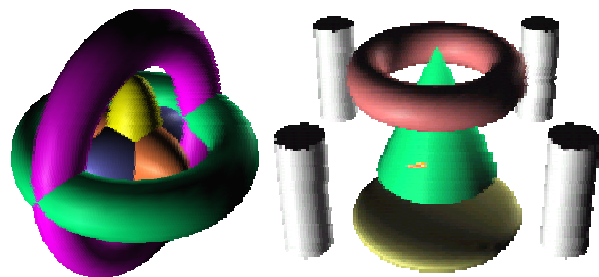


図4 OpenGL を用いた表示例

表2 基本立体の生成コマンド

生成立体	コマンド	主なパラメータ
円錐	Cone	立体名, 半径, 高さ
直方体	Cube	立体名, X,Y,Z 長さ
円柱	Cylinder	立体名, 半径, 高さ
楕円体	Ellipse	立体名, XY 半径, XZ 半径
直線	Line	立体名, 始点座標, 終点座標
多角柱	Prism	立体名, 高さ, 座標点列
多角錐	Pyramid	立体名, 高さ, 座標点列
回転体	Revolve	立体名, 座標点列
球	Sphere	立体名, 半径
円環体	Torus	立体名, 大半径, 小半径
螺旋体 1	Spiral	立体名, 大半径, 小半径, ピッチ
螺旋体 2	Helix	立体名, 座標点列, ピッチ

表3 立体操作コマンド

機能	コマンド	書式または主なパラメータ
立体のコピー	=	複写先立体 = 複写元立体
集合演算	=, +, -, *	立体 3 = 立体 1 (op) 立体 2
アフィン変換	Affine	立体名, 中心, 変換行列
消去	Delete	立体/グループ名
平行移動	Move	立体名, 相対変位
複数コピー	Ncopy	立体名, コピー方向, 数
回転移動	Rotate	立体名, 回転中心, 軸方向
カラー設定	Xcolor	立体名, 表示色(R.G.B.)

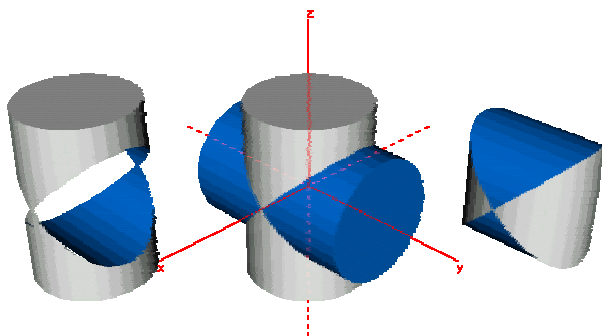


図5 集合演算の例(和, 差, 積集合)

Solid Interpreter のコマンドには、繰り返しや条件分岐などの実行制御をコントロールすることができるコマンドが用意されている。さらに、変数が使用できるため、プログラミングにより立体のパラメータが変更できる「拡張パラメトリック表現」が可能である。たとえば、次の3D プログラムのように、変数により、羽根の枚数を指定してプロペラを生成することができる。通常のパラメトリック表記は、立体のサイズを数値として指定するものであり、基本的な形態を変化させるものではない。

コマンドを組合すことで、複雑な立体を生成することができる。図6に例として、大学生が作った作品を示す。8時間から10時間程度使い方を教えた後に課題として提出されたものである。これら以外にも非常に凝った作品が多くあり驚かされた。これらの立体モデルはグループ構造と軸動作機能を利用して動くように設計されている。

4. 3D プログラミングのための GUI 環境

開発したソフトウェア「Solid Interpreter」は、立体生成

やコマンド生成をサポートする次の5種類のツールを備えている。

- (1) **メイン Window :**
3D プログラミング用のマルチテキストエディタを持ち、実行制御、ファイル管理を行う。
- (2) **Primitive 生成 Window :**
基本立体生成コマンドを GUI 環境で自動的に生成するツールで、他のツールを呼び出す機能も持つ。
- (3) **Command 生成 Window :**
基本立体生成以外のコマンドを生成するためのツールで、色設定などビジュアルに指定することができる。
- (4) **状態表示 Window(Object Inspector) :**
実行時のグループ関係を Tree 表示や、メモリの使用状況を調べたりするためのツールである。
- (5) **グラフィック Window :**
立体を表示するだけでなく、ビットマップ画像の保存やクリップボードへの転送機能を持っており、表示画像の再利用を支援する機能を有する。(この報告書のグラフィックス画像はこの機能を利用してワークプロに貼り付けている。)

これら以外にも、初期設定を行うためのフォームや数値パラメータを入力するためのフォームなど多くのプログラム機能を有している。

これらのツールを使用することで、GUI 環境下でビジュアルに立体を生成したり、コマンドを生成することができる。図7に基本立体を生成するためのツールである「プリミティブ生成 Window」(任意対称断面の螺旋体生成時の表示例)を示す。このツールにより37種類の基本立体を確認しながら生成することができ、さらに、パラメータが多い視線情報も確認しながら設定することができる。



図6 3次元立体の製作例

5 . 数式計算機能と関数表示機能ほか

5.1 数式計算機能

Solid Interpreter Ver.3 では数式計算は四則演算までしかできなかった。これを、三角関数や指数関数の計算まで可能にし、さらにコマンドのパラメータ部分でも数式で表現可能にした。これに伴い、従来採用していたパラメータの区切り記号として、“,”のみを受け付けるように変更した。(従来はスペースまたはコンマ)このため、Ver3 の3D プログラムは実行できない場合がある。しかし、できるだけ互換性を保つようになっている。数式計算機能により、例えば次のような記述が可能になる。

- (1) Cone cc01 16, R*2/(p+1), H*(1+sin(T))
- (2) Repeat P*2+4 { }
- (3) Axis sl01 R*cos(T*3.1415/180)
- (4) Move SolidX R*cos(t), R*sin(t), 0

これにより、従来は複数の直線的動作で近似していた動作で、たとえばエンジンのクランクシャフトやカムの動作などを正確に記述することができる。

5.2 関数表示機能

数式計算ができるようになり、立体のパラメータも数式で表現できるようになった。これに付随させて、数式で頂点を与え、その頂点から立体を生成できる機能を付け加えた。このような機能は、数式表示を行うソフトでは当然のごとく可能であるが、本ソフトウェアでは数式で与えられた曲面や曲線を立体モデルとして構築するところが異なる。CG ソフトでは通常サーフェスモデルとして生成するが、ここではソリッドモデルとして立体が生成される。このため、パラメータには曲面や曲線の情報だけでなく、厚さや半径の情報が加わっている。暫定的ではあるが、“Plate” コマンドとしてスクリプトを構築した。変数として (x, y, z) を使用し、最後の1行または2行がパラメトリックな変化に使用され、パラメトリック部が1行の場合は曲線、2行の場合は曲面とする。Plate コマンドの書式は、

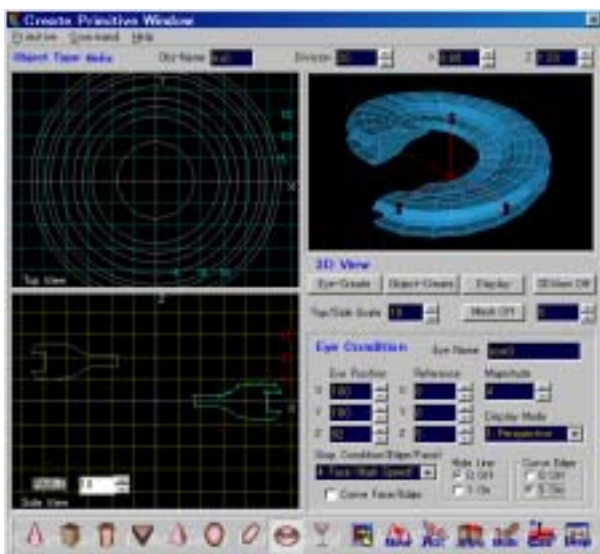


図7 プリミティブ生成 Window

Plate - オプション 立体名 式の数, 厚さ; 式1 . .

である。オプションは “l” が曲線形状, “f” が曲面であり、曲線の場合は、式の数 は 2 ~ 4 でなければならず、曲面の場合は、式の数 は 3 ~ 5 でなければならない。厚さは曲線の場合は線の半径を表し、曲面の場合は面の厚さを意味する。式の計算による (x, y, z) の座標が頂点となり、立体の位相データが生成される。曲線の場合は最後の式がパラメータとなり、曲面の場合は最後の2式がパラメータとなる。以下に曲線(1)と曲面(2)の書式の例を示す。

- (1) Plate -l LBody 4, 0.001; z = 0.4*t;
x = r*cos(t*2); y = r*sin(t*2);
t= -3, 3, 0.10;
- (2) Plate -f FBody 3, 0.01;
z = 0.6*(sin(x*2) * cos(y*3));
x = -1.6, 1.6, 0.1;
y = -1.6, 1.6, 0.1;

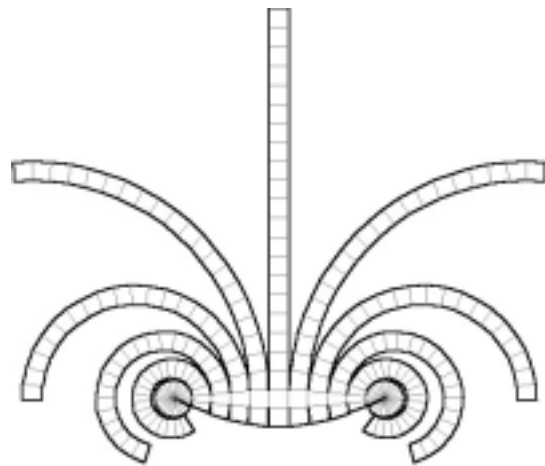
書式(1)は、変数 t が-3 から+3 まで 0.1 ずつ増えながら、角柱状の立体が形成される。書式(2)では、x,y がそれぞれ-1.6 から+1.6 まで 0.1 ずつ増えるように曲面が形成される。図 8(1),(2)にそれぞれの立体を示す。また、図 9 は図 8(2)の立体の厚さを大きくしたものである。

5.3 展開図の生成

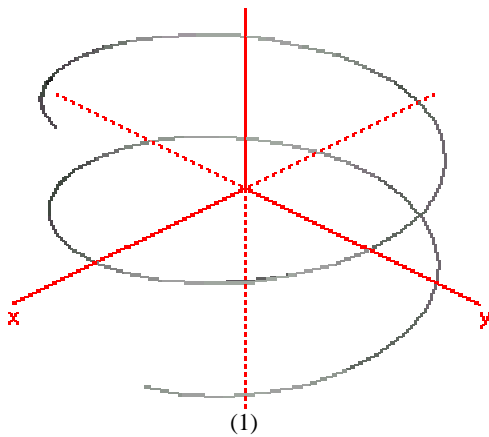
展開図の生成機能は、小中学校からの要望もあり、今後小中学校に本ソフトウェアを普及しようとする場合に重要な機能となると考えられる。展開図の生成方法には幾つかの方法が考えられるが、現時点では2種類の方法を実現している。1つは、単純に面番号順に展開しつなげていく方法で、もう一つは大きい面順に回りの面をつなげながら展開していく方法である。どちらの場合も、面をグループ化して扱い、グループ単位で展開する。Solid Interpreter では、面および稜線をグループ化する構造を取っており、例えば円柱の場合は底面と上面、側面の3グループの面があるという形を取っている。集合演算が行われた場合にも面グループは保存される。たとえば穴をあけた場合には、穴の側面は円柱の側面のグループと同じデータを継承するようになっている。この機能は展開図を生成する上で都合よく機能する。通常、穴の空いた立体などを展開することは、面同士が重なるため困難である。しかし、面がグループ化されているので面のグループ単位で展開することで、この重なりを避けることができる。問題点は、展開図が複数の図形になることである。図 10 は2種類の方法で球を展開した図である。また、図 11 は32面体と穴をあけた直方体の展開図である。基本的にはどんな複雑な立体モデルも展開できる。図 12 は、ロボットの足の部分を展開したものである。5 ~ 6回の集合演算を施した複雑な立体であるが、部品ごとに展開できていることがわかる。展開する方法は簡単で、展開したい立体モデルをリストから選択し展開ボタンを押すことで展開図が生成される。展開された複数の図を、マウスやリストから選択し、移動や回転を施して形を整えてビットマップファイルまたは JPEG ファイルに出力することができる。

5.4 入出力インターフェイス

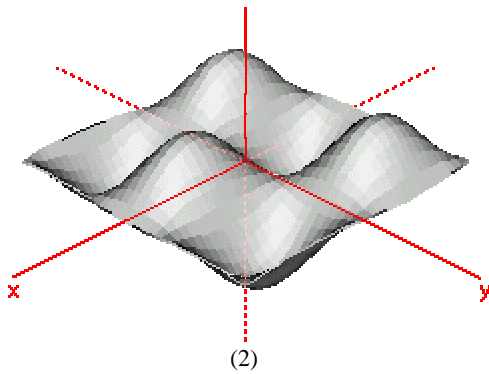
Solid Interpreter の言語体系に入出力機能をつけることを2年前より計画し、USB 接続による入出力回路を設計してきた。現時点でほとんど完成しており、高性能な制御装置としても使用できるようになっている。USB のインターラプトとコントロール転送機能を使い、コンピュータと通信をしながら、3ポート 24 ビットの入出力と6Channel10 ビット分解能の A/D 変換器、2Channel8 ビット分解能の D/A 変換器、さらに2Channel の RS232C インターフェイスをコントロールすることができる。Solid Interpreter からは、わずか3種類のコマンドでこれらの機能を使うことができるようになっている。また、Windows のC言語や VisualBasic などからも制御できるようにしている。図 13 は、USB インターフェイスの写真で、図 14 は接続時の写真である。



(1) 球を単純な方法で展開



(1)



(2)

図 8 曲線と曲面モデルの表示

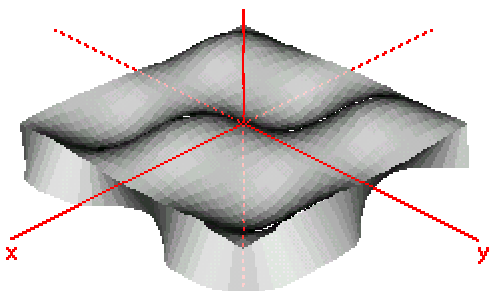
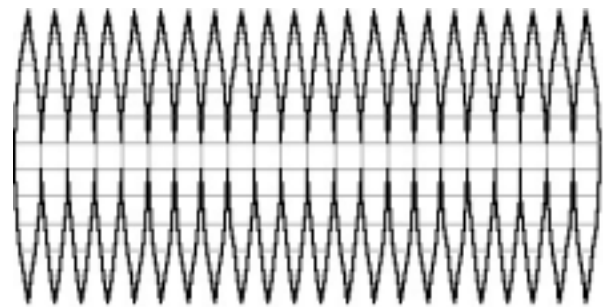
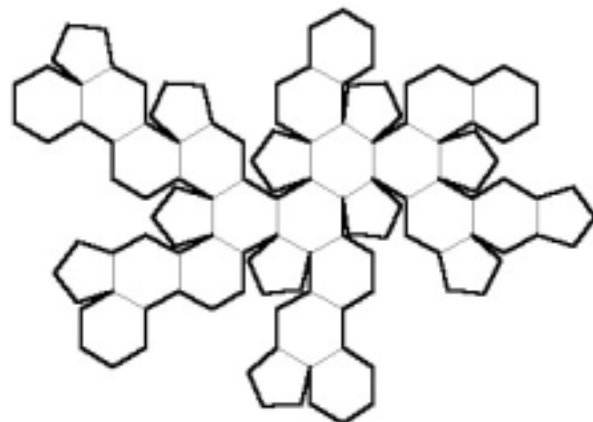


図 9 厚さを大きくした曲面モデル

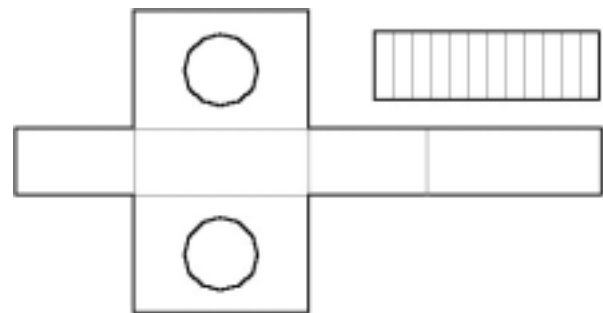


(2) 球を大きい面の回りから展開

図 10 展開図の例 1



(1) サッカーボールの展開図



(2) 穴の空いた直方体の展開図

図 11 展開図の例 2

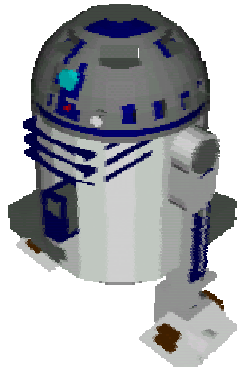


図 14 USB インターフェイスによるおもちゃ制御

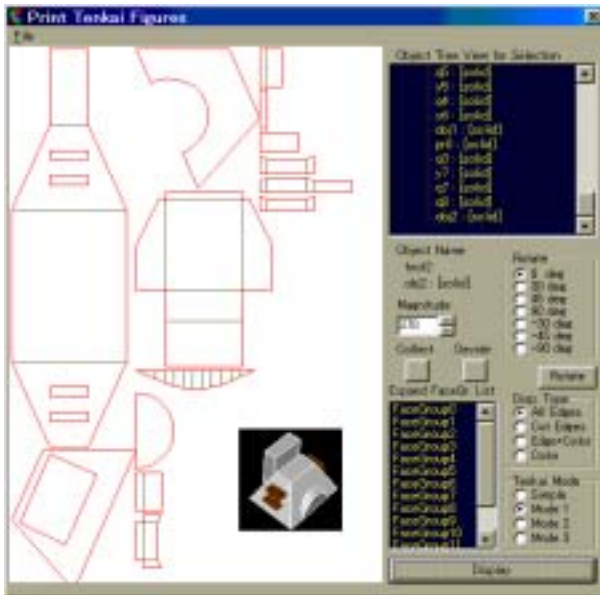


図 12 ロボットの足の部分の展開図



図 13 USB インターフェイス

6. 結論

平成 13 年度に開発したソフトウェアは、ソリッドモデラーの機能の充実とともに、主に上述の第 5 章に示した

- (1) 数式計算機能
- (2) 関数表示機能
- (3) 展開図の生成
- (4) 入出力インターフェイス

である。今後は、これら新機能を盛り込んだ新しいバージョンの「Solid Interpreter Ver.4」の製作と販売開始を目指して開発を進める。また、入出力インターフェイスについても近日中に販売を開始する予定である。

開発に当たっては、ネプラス株式会社に協力をお願いした。特に入出力インターフェイスの開発に協力していただいた。また、開発の成果の一部は下記のホームページで参照することができる。

<http://www.cyber-solid.com/>
<http://www.n-plus.co.jp/>

是非、参照していただきたい。

7. 参考文献

- [1] 山口富士夫：コンピュータディスプレイによる図形処理工学，日刊工業新聞社，pp.6-45 (1981)
- [2] 千代倉弘明：ソリッドモデリング，工業調査会，pp.12-22 (1985)
- [3] 鳥谷浩志・千代倉弘明：3次元CADの基礎と応用，共立出版，pp.16-142 (1991)
- [4] 新津 靖：図学およびコンピュータグラフィックス教育のための3次元立体生成ソフトウェアの開発，私立大学情報教育協会論文集，Vol.2, No.1, pp.7-12 (1999)
- [5] Yasushi Niitsu: "Solid Modeling with Polygon Data of Mathematica", Proc. of 4th Int. Mathematica Symp., pp.369-374 (2001).