

電子工学分野における結晶構造描画表示システムの開発

杉浦 藤虎¹⁾・小澤 功¹⁾・塚本 武彦¹⁾・安藤 浩哉²⁾

電子工学分野における結晶構造の描画表示システムを構築した。従来、原子構造の説明には板書やプリントの配布以外に、手のひらに収まる程度のプラスチックモデルを回して確認させたり、原子模型の写真をプロジェクタに投影したりするなどして行ってきた。しかし、肉眼では見えない立体構造を頭の中にイメージし、3次元的に回転させてその性質を理解させることは容易ではなく、時間も要した。そこで、本研究では3次元の結晶構造を主にマウスによるアニメーションとして表現することを試みた。3Dグラフィックスを比較的簡単に実現するため、あらゆるハードウェアで利用可能な、フリーソフトのOpenGLとその関連ライブラリを利用した。特に、視点や角度の変更をリアルタイムに行えるよう、操作しやすいグラフィカルユーザインターフェース (GUI) を考慮した。それらの機能によって、速やかにかつ容易に、見たい角度からの結晶構造を表示させることができた。単純な基本格子だけでなく、複数個の格子表示やxyz方向の任意の回転、格子フレーム、ボンドの描画、面指数による結晶面の表示などがマウスやキーボードから容易に行える。光源による影や陰面処理なども施してある。またメモリやCPUの使用量も抑えるように考慮しているため、アニメーションの描画は滑らかである。

キーワード

結晶構造、空間格子、基本単位格子、3Dグラフィックス、リアルタイム処理、OpenGL

1. はじめに

電子工学分野では原子や電子などの肉眼で見えない粒子を取り扱う機会が多い。そのため、教科書などの挿し絵は定性的理解に対する、またグラフや実験データなどは定量的理解に対する助けとして非常に有効である。さらに、新機能材料の開発や材料固有の性質の探求に際し、材料の微視的結晶構造を理解することは、そこから得られる材料の巨視的特性を知る上で非常に重要である。

結晶構造は原子がエネルギー的に安定となるよう規則正しく並んだ結果である。それ故、ある方向から眺めると非常に美しい幾何学模様が現れ、芸術的でさえある。特に半導体や誘電体などの剛性、劈開性や異方性などは結晶構造の方向と密接に関わっている。従来、原子構造の説明には板書やプリントの配布以外に、手のひらに収まる程度のプラスチックモデルを回して確認させたり、写真をプロジェクタに投影したりするなどして行ってきた。しかし、立体構造を頭の中にイメージし、3次元的に回転させてその性質を理解することは容易ではなく、時間もかかる。講義時間内での説明だけでは理解が十分

ではないため、学生は実際に自分でじっくりと結晶を眺め、眺める角度を変えて考えることが大切になる。これまで、結晶構造や分子構造を描画・可視化するアプリケーションは市販のソフトウェアも含めていくつか存在する。しかし、それらは化学の研究用の本格的な分子軌道計算や分子軌道・電子密度描画が可能な高機能なものが多く、電子工学の基礎として講義の中で利用することは適切でない。また、高性能なワークステーションや整備されたハードウェアが要求されたりして、必ずしも使い勝手の良いものではない。さらに原子の大きさを最隣接原子と接するまで任意に変更したり、結晶面や方向ベクトルなどが表示されたりする機能が付加されたソフトウェアは見当たらない。

そこで本研究では講義用補助教材として、上記の機能を有し、結晶格子を複数個描画したときにも表示速度が遅くならないよう高速性と主にマウスで扱えるよう操作性を重視した電子工学分野に特化した結晶構造の描画表示システムを構築することを目指した。

2. システムの基本仕様

3次元の結晶構造をアニメーションとして表現するため、3Dコンピュータグラフィックス (CG) 手法を用いた。3D-CGを作成する上で、満たすべき設計方針として以

¹⁾ 豊田工業高等専門学校 電気・電子システム工学科

²⁾ 豊田工業高等専門学校 情報工学科

下の項目を考慮した。

1. 任意の視点から描画、表示できること
2. 任意の格子構造を複数個描画、表示できること
3. 任意の結晶面指数に対応した平面を描画、表示できること
4. 高速に描画し、滑らかなリアルタイム表示ができること
5. 操作性のよいグラフィカルユーザインターフェース (GUI) であること

上記仕様を実現するために、本研究ではC/C++言語を用いてプログラムの開発を行った。プログラミング開発の環境として、主としてあらゆるハードウェアで利用可能な3D-CG用のOpenGL、GLUT、GLUIというフリーのライブラリを利用した。描画にはOpenGL、ウィンドウ・システムの操作にはGLUT、インターフェイスにはGLUIを用いた。ただし、利用時の利便性を考え、マイクロソフト系のプログラム開発環境となるWindows上で開発を行った。本研究ではマイクロソフト社のオブジェクト指向言語であるビジュアルC++とOpenGLを組み合わせて、3D-CGを高速かつ滑らかに表示することを試みた。作成したプログラムは教員が講義時に補助教材として利用する他、研究室のホームページ (HP) 上におき、学生各自でダウンロードして利用する。以下に使用した開発環境を簡単に述べる。

(1) OpenGL

OpenGLとは、OSに依存しない (オープンな) グラフィックスライブラリのことで、極めて移植性が高く、多くのシステムで利用できる。OpenGLはレンダリング処理のみを行い、ベンダに中立なAPIとして2Dと3Dのグラフィックス関数を提供する。そのため、プラットフォームに依存せず、ウィンドウや入出力などを制御するコマンドは含まれていない。しかし、それらの点に関しては関連ライブラリであるGLUTなどで解決できるよう配慮されている。OpenGL関連ライブラリもまた、システムには依存しないオープンな仕組みとなっている。

(2) GLUT

OpenGLは機能がシステムから完全に分離されているため、直接、システムとの連携ができない。そのため、システムと連携するためのライブラリがいくつか用意されており、中でもよく利用されるライブラリがGLUT (OpenGL Utility Toolkit) である。上述のように、このライブラリはシステムから独立しているためシステムの機能を最大限に発揮させることは難しく、また、システムのAPIを使う方法に比べ、実行できることはかなり制限される。しかし、多くのシステムでGLUTがサポートされており、OpenGLとGLUTを組み合わせてシステムに依存しないソースコードを記述できる利点は大きく、また比較的簡単なソースコードで実現できる。

(3) GLUI

GLUI (GLUT-based C++ User Interface library) は描画時の視点やオブジェクトの大きさなどを動的に操作するためのGUIライブラリで、C++言語で記述されている。「コマンドボタン」、「チェックボックス」、「ラジオボタン」、「スピンボタン」などのWindowsで見慣れた関数も用意されている。また、視点や角度の変更に対してマウスで操作できる「トラックボール」も比較的容易に利用できる。GLUIの優れた長所は、シンプルなプログラミングインターフェイスであり、容易にさまざまなGUIを配置することができる。

(4) オブジェクト指向

オブジェクト指向とは、ソフトウェアの設計や開発において操作手順よりも操作対象に重点を置く考え方である。個々の操作対象に対して固有の操作方法を設定することで、その内部動作の詳細を隠蔽し、利用しやすくなる。複雑かつ大規模なソフトウェアの作成と管理を高効率化する上で重要な役割を果たす。

3. 結晶格子の描画と表示

本システムでは任意の結晶構造を描けることに重点を置いた。そのため、原子の位置座標などのデータをファイルから読み込む方法を採用し、結晶構造に依存しないようにした。ファイルからの読み込みから描画までのデータの流れを図1に示す。オブジェクト指向のプログラミングでは、各オブジェクトが担当する役割をはっきりさせ、どのような機能やデータを持たせるかを設計する必要がある。

図2に「データ構造ファイル」の例を示す。データ読込では、ファイルの内容を読み込み、「データ構造解析」に1行ずつ渡す。例えば、それが頂点座標だとすれば、「頂

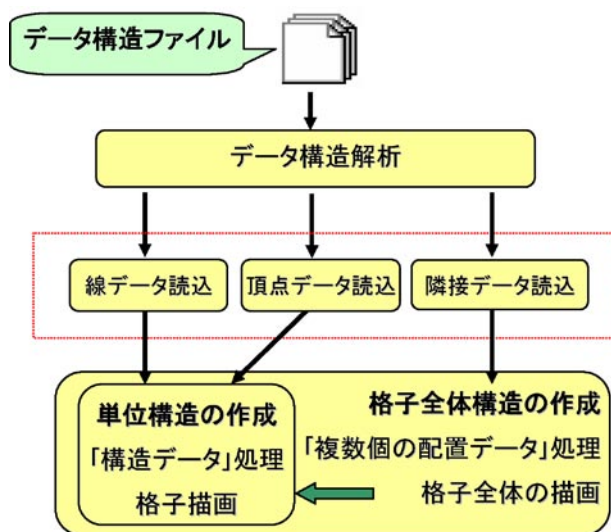


図1 任意の結晶格子の描画

```

://単純立方格子
; //ヘッダ開始
vertex2://頂点の種類の数
line2: //線の種類の数
@; //ヘッダ終了
://本文開始
://頂点
[vertex0]8://立方体の角
-1.0, -1.0, -1.0;
1.0, -1.0, -1.0;
1.0, 1.0, -1.0;
-1.0, 1.0, -1.0;
-1.0, -1.0, 1.0;
1.0, -1.0, 1.0;
1.0, 1.0, 1.0;
-1.0, 1.0, 1.0;
[vertex1]1://立方体の中心
0.0, 0.0, 0.0;
://線
[line0]12://フレーム
0: 0, 0: 1;
0: 1, 0: 2;
0: 2, 0: 3;
0: 3, 0: 0;
0: 0, 0: 4;
0: 1, 0: 5;
0: 2, 0: 6;
0: 3, 0: 7;
0: 4, 0: 5;
0: 5, 0: 6;
0: 6, 0: 7;
0: 7, 0: 4;
[line1]8://結合
1: 0, 0: 0;
1: 0, 0: 1;
1: 0, 0: 2;
1: 0, 0: 3;
1: 0, 0: 4;
1: 0, 0: 5;
1: 0, 0: 6;
1: 0, 0: 7;
//基本並進ベクトル
[neighbor]3;
2.0, 0.0, 0.0;
0.0, 2.0, 0.0;
0.0, 0.0, 2.0;

```

図2 「データ構造ファイル」の例（単純立方格子）

点データ読込」に渡される。頂点データ読込では、読み込んだ文字データを数値データに変換し、「単位構造の作成」に必要な数だけ動的確保された、メンバ変数に格納する。このようにして全てのデータは「構造データ」に格納されていく。動的確保されることにより「構造データ」には任意のデータを任意の数だけ入れることができる。その結果、任意の構造を表現することが可能となる。

基本格子全体を描画する場合には、「格子全体構造の作成」が実行される。「格子全体の描画」が呼び出されると、「隣接データ」に基づき「複数個の配置データ」を生成する。それらのデータを利用し、「構造データ」

および「格子描画」によって、基本格子全体の描画を行う。このようにオブジェクト指向を使えば、各オブジェクトの組み合わせとしてプログラム全体を作ることができる。ここで使用したファイルのデータには、「原子の頂点位置座標」や「どの頂点とどの頂点をつなぐのか」などの具体的データが記録され、拡張性を考慮してコメントなどが付けてある。あらかじめ、表計算ソフト等で位置座標を計算し、それを利用しやすい「CSV形式」等で保存すれば、どのような格子構造でも、データの作成と編集を簡単に行うことができる。作成した結晶構造の描画表示例を図3に示す。

3.1 任意の角度からの表示

(1) 視点位置の変更

3次元構造を任意の角度から表示することは、結晶構造の幾何学的特徴を理解する上で最も重要である。これをディスプレイなどの平面上で実現させるためには回転操作が必要となる。物体を回転させるにはマウスを利用するのが便利である。マウスが描画エリア内であれば、その位置情報を基に回転角を計算し、マウスが動くときにそれに応じて回転するようにした。視点の変更は、これ以外にも図4のようにトラックボールで行うことも可能である。マウスでトラックボールをドラッグすると、それに応じてオブジェクトは回転する。さらにトラックボールが選択されている状態で矢印キーを押すと、その表示状態から視線に垂直な平面内でオブジェクトを回転できるようにした。

(2) 適切な大きさで表示

回転操作の際、表示物体がディスプレイ画面や表示ウィンドウからはみ出さないようにしたい。また描画エリアを有効に利用して、なるべく大きく表示すべきである。そこで図5(a)に示された投視射影の視体積に、表示させたい構造物が全て収まるように考慮した。視体積の境界面から成る立体は視錘台と呼ばれるが、この視錘台にちょうど良く収めるためには、図5(a)に示すように、表示させたい構造物の最大回転半径よりわずかに大きい球を考え、その球を視錘台に内接するような条件の下で変換を行えば実現できる。どのような構造あるいは個数でも、オブジェクトはこの球内に収まるので、視点の変更をしてもオブジェクト全体を観察することが可能になる。

その条件を満たす関係式は以下のようにして導出される。図5(b)は物体の表示サイズを決定する過程の説明図である。考える球は視錘台に内接するので、図5(b)中の y_1 は表示ウィンドウ(図5(a)のハッチ部分)の縦あるいは横の長さの短い方を表している。図5(a)においては縦の長さの方が短い。図5(b)中で、 $\triangle OAB$ と $\triangle OCD$ は相似になるので①式が成立する。さらに、 $\triangle OCD$ と $\triangle OQP$ は相似なので②式が成立する。①式を

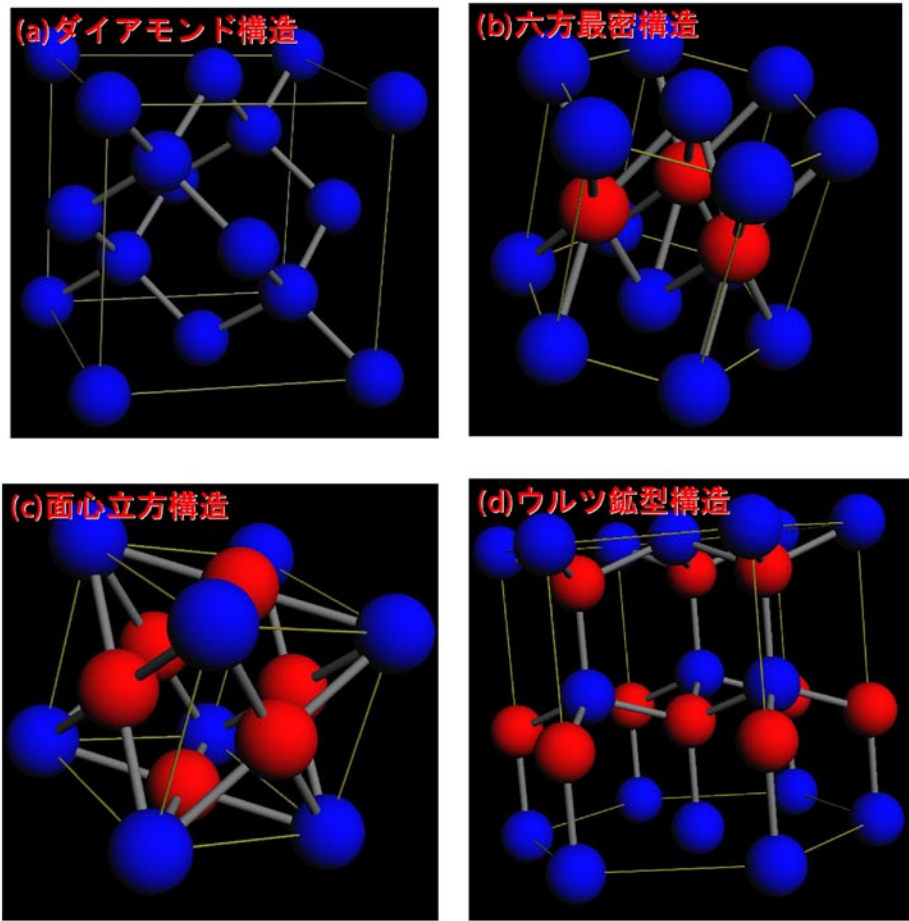


図3 描画表示例
(a) ダイヤモンド構造、(b) 六方最密構造、(c) 面心立方構造、(d) ウルツ鉱型構造

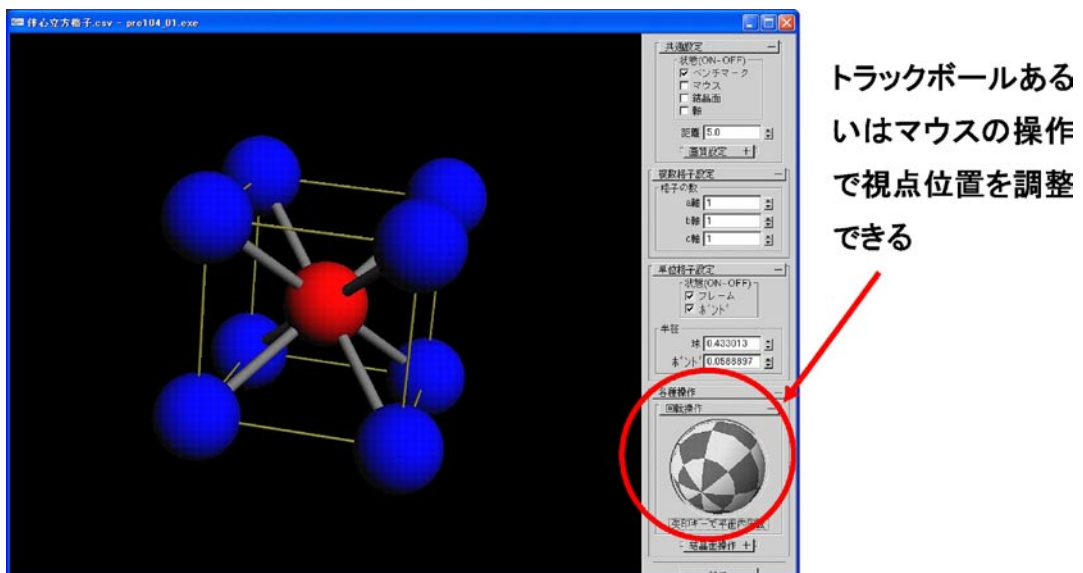


図4 視点位置の変更操作

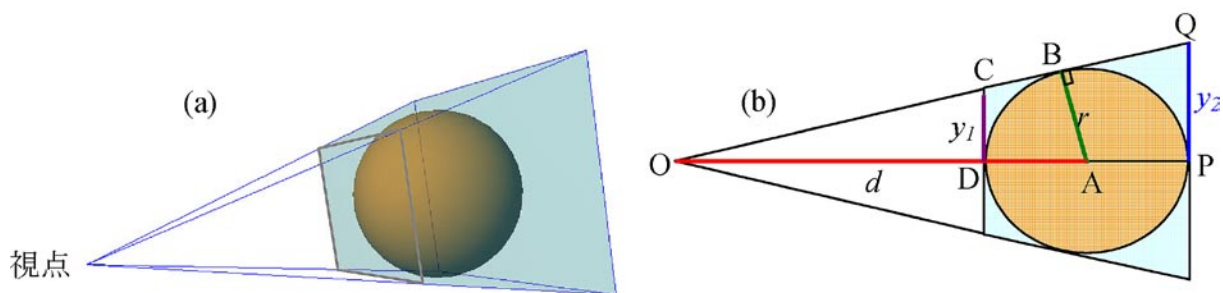


図5 (a) 視錐台（見える範囲：水色）と視錐台に内接する球（茶色）、(b) 表示サイズの説明

変形して y_1 を求めると③式、②式を変形して y_2 を求めると④式が得られる。以上の式を用いて射影変換すれば、表示ウィンドウのサイズに一致した大きさで見ることができる。

$$\frac{r}{d} = \frac{y_1}{\sqrt{(d-r)^2 + y_1^2}} \dots \text{①}$$

$$\frac{y_1}{y_2} = \frac{d-r}{d+r} \dots \text{②}$$

$$y_1 = r \sqrt{\frac{d-r}{d+r}} \dots \text{③}$$

$$y_2 = y_1 \times \frac{d+r}{d-r} \dots \text{④}$$

また、図6に示すように結晶中の原子（剛体球）の半径はGUIパネルから最隣接原子と接するまで任意に変えることができる。その結果、最隣接原子と接したときの原子が占める体積の単位格子体積に対する割合、すなわち、充填率を求めることができる。

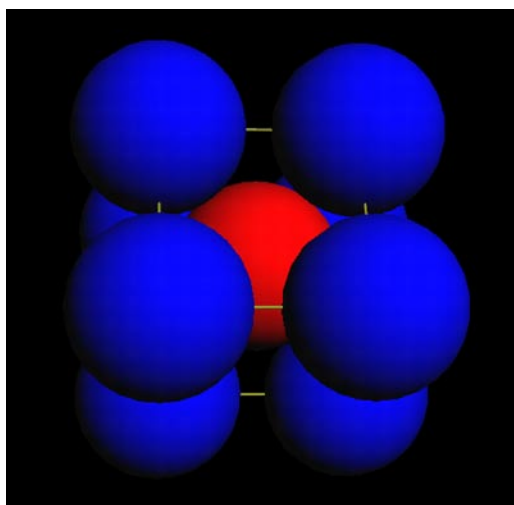


図6 体心立方格子が最隣接原子と接したとき

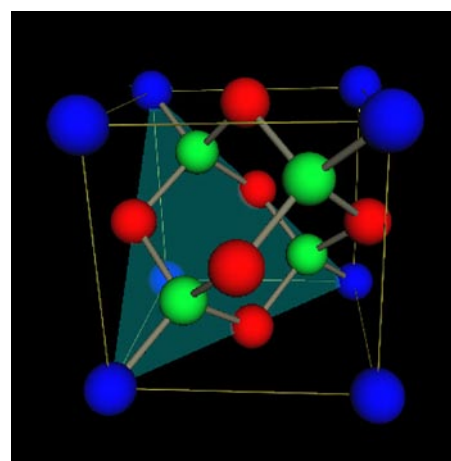


図7 結晶面の表示例

3.2 結晶面表示

結晶内に規則正しく並んだ原子からなる面の、どの結晶面かを特定するためのミラー指数は、結晶構造固有の座標系で表現されるため、例えば立方晶と六方晶では表現が異なる。そこで結晶構造固有の座標系を標準の直交座標系に変換した後で処理することにし、構造に依存しないアルゴリズムを利用できるようにした。具体的には「データ構造ファイル」に表示したい結晶構造の基本ベクトルを直交座標系で記述することで、結晶構造固有の面指数を標準の座標系に変換し、図7に示すような結晶面を表現した。

3.3 複数個の描画

複数個の基本格子構造を描くことで結晶の周期性を幾何学的に観察することができる。結晶の周期性はベクトル手法を用いて表現した。すなわち、結晶の基本並進ベクトルを用いて複数個の描画処理を行った。基本並進ベクトルを $\mathbf{a}, \mathbf{b}, \mathbf{c}$ とし、基準の単位格子の中心を \mathbf{r} とすると、全ての単位格子の中心 \mathbf{r}' は次式で表される。

$$\mathbf{r}' = \mathbf{r} + n_1 \mathbf{a} + n_2 \mathbf{b} + n_3 \mathbf{c}$$

ただし、 n_1, n_2, n_3 は任意の整数で、それぞれを独立に指定することにより描画する個数を結晶軸ごとに可変できる。簡単のため、正六角形の基本格子を2次元平面状

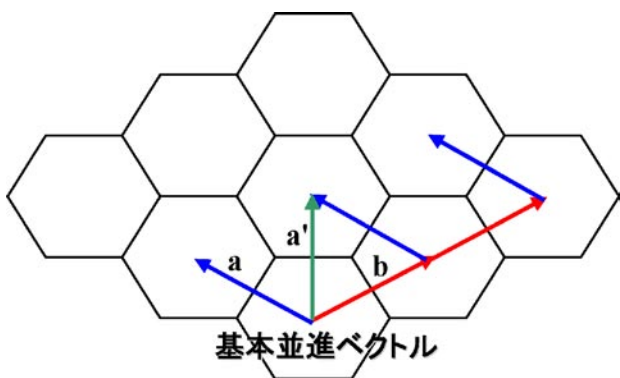


図8 基本並進ベクトルと複数の基本格子との関係

に図8のように配置することを考える。例えば、図8に示すような二つの基本並進ベクトル \mathbf{a} , \mathbf{b} (基本並進ベクトルは他にも \mathbf{a}' のようにいくつか考えられるが、特に指定はない) を考え、ベクトルが示す位置に基本格子の重心を配置すれば、2次元平面上に隙間なく六角形を埋めることができる。同様の手法を3次元空間に拡張して、基本格子を3次元空間で任意の個数だけ描画する。

また逆に、任意の構造が描けることと、任意の個数で描けることを応用すれば、与えられたデータが空間格子として実現可能な構造かどうかを確かめることもできる。基本並進ベクトルは頂点情報から算出することは容易でないため、ファイルから直接読み込むようにしている。

3.4 操作環境

快適に操作するためには、いろいろなパラメータを動的に操作できる必要がある。パラメータの変更は画面上のボタン操作で行えるよう、特に使いやすい操作性を重

視した。したがって、ほとんどの操作をマウスで行うことができる。本研究で使用したGUIパネルを図10に示す。

3.5 高速化

実際に3Dグラフィックスを滑らかに動かすためには高性能なグラフィックエンジンを搭載することが最も簡単であるが、その分費用がかかる。そこで、ハードウェア的な改良を加えずにプログラムの効率化を図り、ソフトウェア的に高速化することを試みた。高速化手法として、表示するポリゴン数を表示に支障のない程度に減らすことで対処した。ポリゴン数はパーツ(球や円柱など)を表示した時の大きさ(ピクセル数)によって可変できるようにした。相対的な大きさは、表示エリア1辺のピクセル数と相対半径(構造物の最大回転半径に対するパーツの半径)との積で算出した。

予備実験において、相対的な大きさに基づいてポリゴン数を線形に変化させたとき、処理速度は線形に変化しなかったため、図11中に示した式を用いて、指数関数的にある一定値に近づくようにした。その結果、半径の全領域で適切な分割数となり、高画質と高速性を両立することができた。全体的に画質を向上させたいときには、最大分割数 M を手動で可変できるようにした。 M や相対半径 (r/R) などのパラメータが変更されたとき、分割数は自動計算されるようにした。

4. 本システムの評価

本システムの教育利用に対する有効性と操作性などについて、本校の4、5年生に利用してもらい、16名からアンケートの回答を得た。その結果を図12に示す。本システムの講義時の補助教材の有効性については、回答

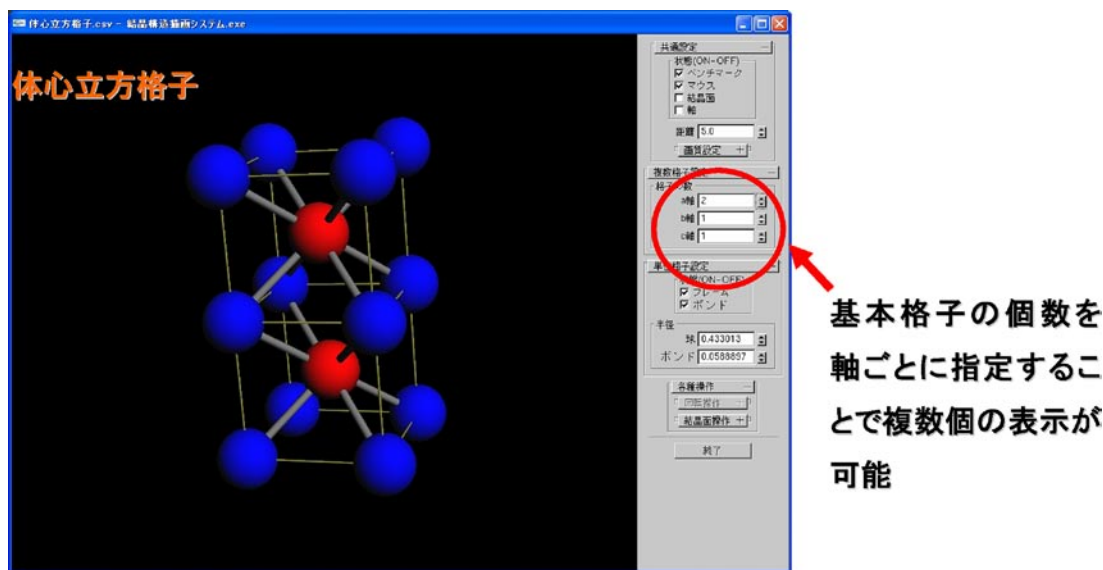


図9 複数の基本格子の描画表示

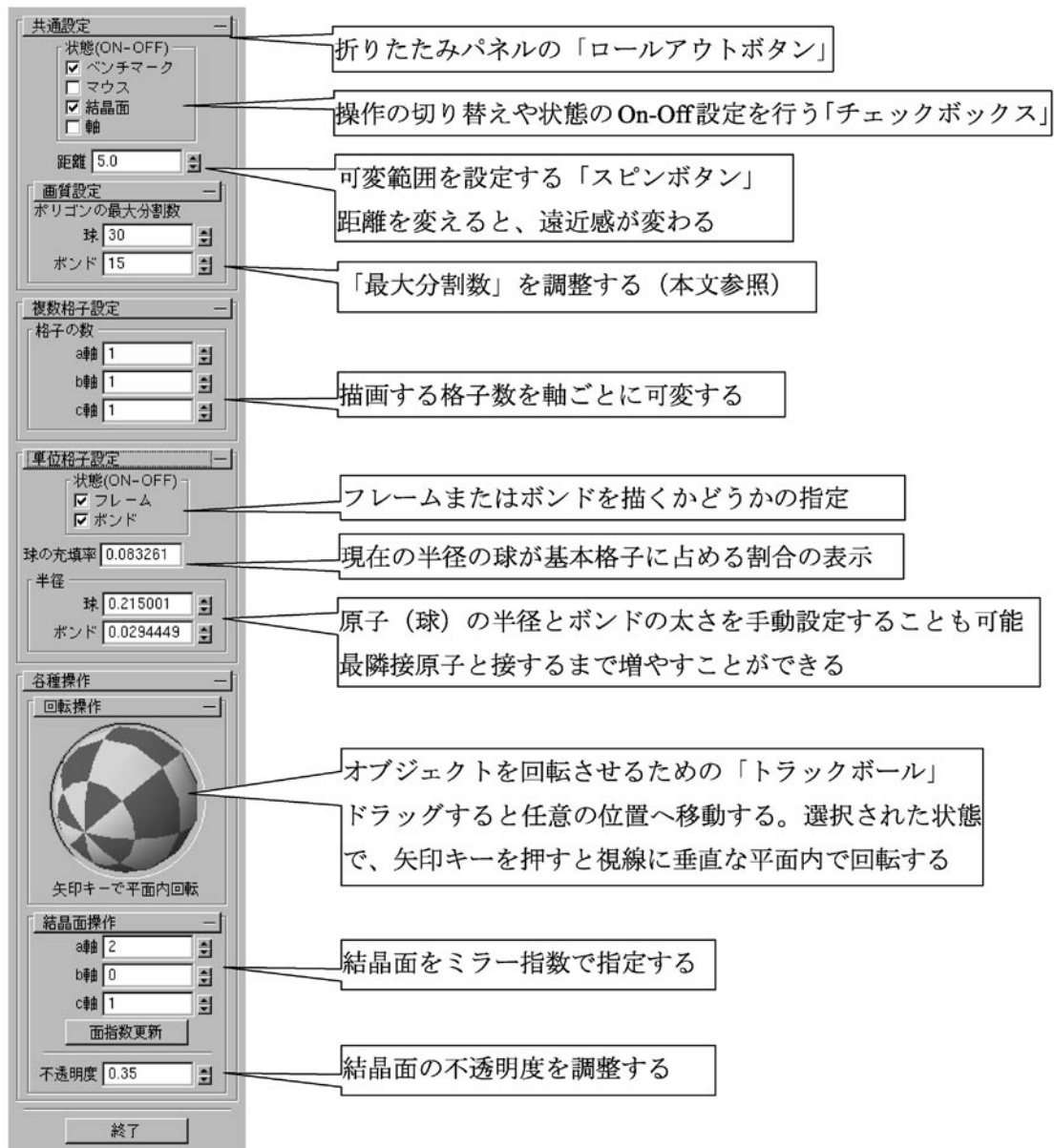


図10 GUI パネル

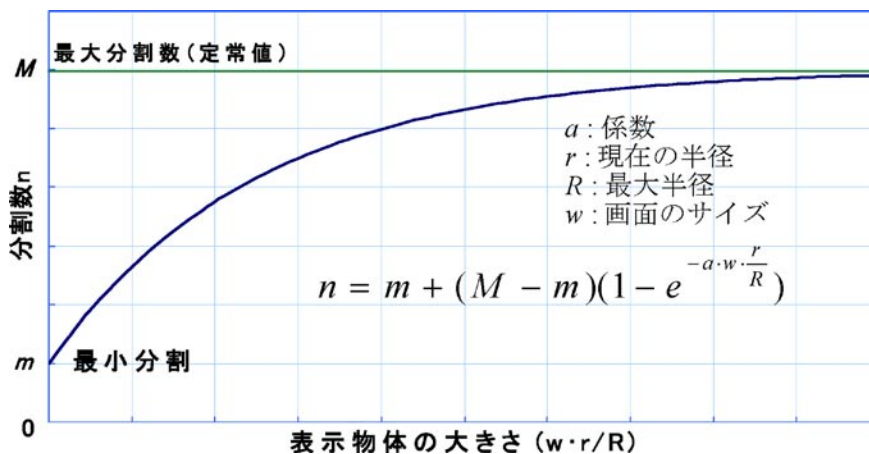


図11 ポリゴンの分割数と表示物体の大きさとの関係

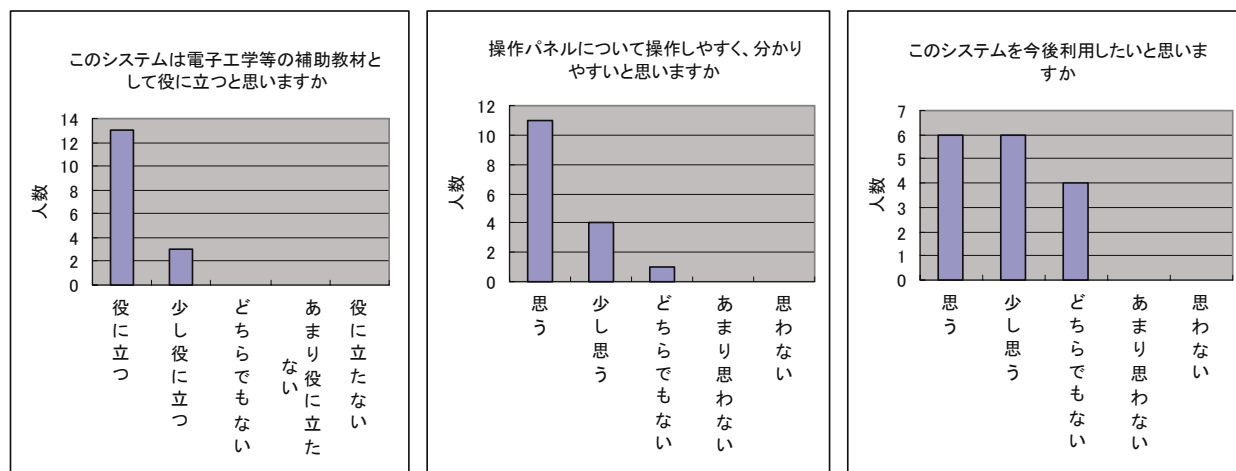


図12 本システムに対するアンケート結果

者全員から「役に立つ」、もしくは「少し役に立つ」の評価を得た。操作パネルに関する使いやすさについても、「どちらでもない」を回答した1名を除き、ほぼ全員から好評を得た。また、今後このシステムを利用したいと思うかとの問いに対して、回答者の3/4の学生から「思う」「少し利用したい、時々利用したい」との評価が得られた。以上の結果を踏まえ、主に教員の講義時の補助教材として利用しながら、webを通して学生個人にも配布する予定である。なお、自由記述欄に寄せられた改良や追加すべき点の意見として、

- ・ Web (インターネット) を通して利用できるとよい (2名)
- ・ 欠陥とかが再現できるとおもしろい
- ・ マウスが少しずれるだけで過敏に反応してしまうので微調整機能が欲しい
- ・ 人が選択できるようなメニューがあればよい
- ・ 断面図が見られるようにして欲しい
- ・ 下位学年の化学の授業などでも利用できるよう、表示できる構造体の種類を増やすとよい
- ・ 原子間の長さや角度も表示されるとよい
- ・ 格子の表面も表示して、立体表示できるとよい

などがあつた。今後は、上記意見の中にもあつたように、下位学年の講義時に利用してもらうことを含め、寄せられた意見や感想を参考にした上で、改良を加えていきたいと考えている。

5. まとめ

電子工学や半導体などの固体物理分野の講義で扱う結晶構造は、電場や磁場と同様に、肉眼では見えない抽象的な存在である。しかし、材料の微視的構造は巨視的材料の性質を決定する重要な要素であり、その正確な理解は電子デバイスの材料開発や応用に役立つ。本研究で開発した3Dグラフィックスツールは、上記分野の講義時

の補助教材として有益なツールになるものと思われる。一般に、頭の中で原子構造を立体的にイメージし、3次元的に回転させてその性質を理解することは容易ではないが、講義時に本ツールを用いて、広いスクリーン上に投影し、マウスなどで簡単に眺める方向を変えながら説明することができれば、教育的効果は非常に高いと考えられる。また、原子の大きさを最隣接原子と接するまで変えることができるため、実際の材料の物理パラメータから、結晶密度や充填率などが解析でき、手計算により求めた結果と比較することもできる。具体的な数値を扱い、定量化することにより概念の理解の手助けとなると考えられる。今後は、学生からの評価に基づきプログラムの追加や改良などを適宜行うと共に、web上から実行できるように発展させ、自習効果を調査することを考えている。このツールが化学や物理、科学に対する学生の興味を引き出すための一助になればと期待している。

参考文献

- [1] 赤崎 勇：Ⅲ族窒化物半導体、(1999) 培風館
- [2] 安居院猛、関根証明：入門OpenGLグラフィックス、(2001) 森北出版
- [3] C. Kittel: Introduction to Solid State Physics, (John Wiley & Sons, New York, 1986) 6th ed., Chap. 1.
- [4] E. Lengyel, 狩野智英：ゲームプログラミングのための3Dグラフィックス数学、(2002) ポーンデジタル
- [5] GLUI Reference: <http://ktm11.eng.shizuoka.ac.jp/glui/glui.html>
- [6] 松下康之：Mesa3Dで始めるOpenGLプログラミング第1回～15回、Linux Japan (1999～2000)
- [7] OpenGL入門：<http://wisdom.sakura.ne.jp/system/opengl/index.html>
- [8] OpenGL入門講座：<http://miso.ice.ous.ac.jp/~oka/opengl/>
- [9] OpenGL入門 - MesaGLとGLUTによる3次元グラフィックスへの第一歩 -：<http://www.edsys.center.nitech.ac.jp/manual/ogl/>

[10] The OpenGL Utility Toolkit (GLUT) Programming Interface API Version 3: <http://gimlay.org/~andoh/opengl/glut/index.html>



すぎやま たかし
杉浦 隆虎

1963年11月29日生まれ。84年3月豊田高専電気工学科卒業。88年3月三重大学大学院工学研究科修士課程電子工学専攻修了。同年4月豊田高専電気工学科（現電気・電子システム工学科）助手。講師を経て、97年助教授、現在に至る。主として、化合物半導体の分光学的研究に従事。博士（工学）。電子情報通信学会、応用物理学会などの会員。



こざわ いさお
小澤 功

1984年4月30日生まれ。05年現在豊田高専電気・電子システム工学科在学中。主として、コンピュータグラフィクスに関する研究に従事。



つかもと たけひこ
塚本 武彦

1961年6月17日生まれ。82年3月豊田高専電気工学科卒業。89年3月豊橋技術科学大学大学院工学研究科博士後期課程システム情報工学専攻修了。同年4月同大学助手。90年4月豊田高専電気工学科（現電気・電子システム工学科）助手。講師を経て、93年助教授、現在に至る。主として、Bi2212超電導体の研究に従事。工学博士。電気学会、応用物理学会などの会員。



あんどう ひろや
安藤 浩哉

1964年6月22日生まれ。87年3月名古屋大学理学部物理学科卒業。89年3月同大学大学院工学研究科博士課程前期電子工学専攻修了。同年4月豊田高専情報工学科助手。講師を経て、99年助教授、現在に至る。主として、電波天文学、超電導エレクトロニクスの研究に従事。博士（工学）。電子情報通信学会、応用物理学会などの会員。

Development of the Drawing and Viewing System of the Crystal Structures for Electronic Engineering

Touko Sugiura¹⁾ · Isao Kozawa¹⁾ · Takehiko Tsukamoto¹⁾ · Hiroya Andoh²⁾

We tried to develop the animation tool which is able to display the 3-dimensional (3D) crystal structures. In order to realize the 3D graphics, we used the OpenGL graphics library of free software. It can be worked on any hardware. Especially, we took account of the graphical user interface (GUI) in order to enable the change of a viewpoint and an angle mainly by mouse operation on real time. We can display the typical crystal structure with an arbitral angle speedy and easily. It can be showed the plural lattice structures, arbitral rotations of the xyz direction, a lattice frame, and a bond and the indices of the crystal plane etc easily with a mouse or a keyboard.

Keywords

crystal structure, space lattice, primitive cell, 3D graphics, real time processing, OpenGL

¹⁾ Department of Electrical and Electronic Engineering, Toyota College of Technology

²⁾ Department of Information and Science Engineering, Toyota College of Technology