

2L-2

幾何学習支援ツールにおける 立体図形の展開機能

長谷容子

日本アイ・ビー・エム株式会社 東京基礎研究所

1. はじめに

小・中学生の展開図の学習を支援する教育用ソフトウェアには、(1)学習者が展開図そのものをコンピュータに入力し、その展開図から正しく立体図形が組み上がるかをコンピュータが判断し、その正誤を学習者に返す方式と、(2)対象とする立体図形があらかじめ与えられ、それを学習者が展開していくことにより、展開図を得る方式がある。しかしながら、これまでのところ、どちらの方式においても扱える立体図形に限られており、ユーザである学習者あるいは教材として用いる先生の応用範囲は広いとは言えない。そこで我々は、(2)の方式において、あらかじめ与えられた立体図形のみでなく、それらの立体図形に対して切断や投影などを行う一連の幾何学習操作の1つとして、切断後の立体図形に対しても学習者が自由に立体図形を展開したり、組み立てたりできるようにしたいと考えた。(図1)

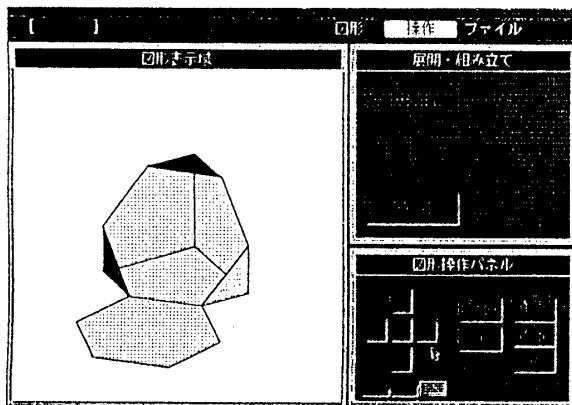


図1. 任意の立体図形の展開
(立方体の一部を切断後の立体の例)

Learning 3D Geometry Through Opening and Closing Faces of Polyhedron

Yohko Hase

IBM Research, Tokyo Research Laboratory

1623-14 Shimotsuruma, Yamato, Kanagawa 242, J

本論文では、任意の立体図形（ただし、平面で構成された凸面体）に対する展開・組み立て機能を実現するための基本データ構造とアルゴリズムについて概要を述べる。

2. 展開・組み立てのための基本データ構造

立体図形は、頂点、辺、面という要素から成っており、図形をコンピュータ上に表示するためには、これらの座標値である幾何学データや、辺・面の色や太さなど豊かなグラフィック表示をするための描画データの他に、各要素間の関係を表現する位相データ（接続情報）が必要である。展開・組み立てにおいては、面が開いたり閉じたりすることにより、一本の辺を境にしてつながっていた他の面との接続が切れたり、逆に離れていた面とつながったりする。そこで、こうした辺を介した面と面との接続関係の探索及びアップデートを効率良く行うために、各面を先頭にしてその面を構成する辺へのポインターと隣合う面へのポインターを含んだ隣接リストを用いる。その際、面間の角度も情報に含める。

3. 展開のアルゴリズム

ユーザが立体図形を展開していく方法として、できるだけ現実の感覚に近く、また、自由に色々な展開の仕方を経験できるように、立体中で切る辺をユーザに直接マウスで指示させ、面を開く形式を考える。

この展開のプロセスは以下のようなものである。

- (1)ユーザが切る辺を指示する。
コンピュータは、
- (2)指示された辺を切ることができるかを判断する。
もし、切れない場合には、(1)に戻る。
- (3)(2)で新しく辺が一本切られたことによって開け

る。もし、ない場合には、(1)に戻る。

(4)面が開いた図形を表示する。

(5)完全に立体が展開されきったかどうかを調べる。

もし、まだ開くことができる面が残っている時には(1)に戻る。

(2)~(5)を行うアルゴリズムは次のようである。

まず、(2)については、ユーザが指示した辺を切ることによって面が立体から離れてしまわないように位相データをチェックする。すなわち、面と面の接続関係を面をnode、辺をedgeとしたグラフで表し、ユーザによって指示された辺を切ったと仮定して、深さ優先探索 (Depth-First Search (DFS))を行い、全てのnodeを訪れることができるかどうかを調べる。もし、全てのnodeを無事に辿れば、その辺を切るものとして位相データをアップデートする。次に(3)については、立体の面が開くときには必ず面の回転の軸となる辺があることを利用する。この辺はグラフ中でbridgeに対応しており、ユーザが辺を切ったことによって新しくbridgeができたかどうかを探せば良い。これはグラフのcut-setに関する問題であり、DFSにおいてvisitorが訪れた順にnodeに番号を付け、同じnodeに再び出くわした際にその番号を持ち帰り、その後の訪問先の番号と比べることにより解くことができ[1]、(2)のための探索時に兼ねて行うことができる。また、ユーザが新しく切った辺は一本であっても、同時に複数のbridgeができることがあるので、全ての新しくできたbridgeを拾い出すことが必要である。(4)ではまず、各々のbridgeを軸にして開く(回転する)面を調べる。これは、あるbridgeを境にして立体を二つの部分に分け、一方に含まれる面全てを求めることであり、再びDFSによって求められる。回転すべき面について座標変換し、幾何学データをアップデートしてコンピュータ上に表示する。この作業を(3)で求めた全てのbridgeについて行う。一度面の回転の軸になった辺は、次章で述べる組み立てのプロセスによって逆方向に回転するときの軸にならない限り、再び軸になることはない。そこで、一度回転の軸となった辺であることを辺のデータに保持しておく。

(1)~(4)のプロセスが繰り返されて、3次元の立体が2次元の展開図に完全に広げられた状態においては、立体を構成する辺は必ず切られたか、1度回転の軸になったかのどちらかである。そこで、立体中の辺のデータを調べて、それら以外の辺があるか

どうかを調べれば(5)が行える。

4. 組み立てのアルゴリズム

組み立てにおいては、開いている面を閉じて組み立てるために、直接ユーザから折る辺を指示してもらう形式を考える。

ユーザから折るように指示された辺が、3章で述べた展開の際に1度回転の軸となった辺であれば、折ることができる。故にその辺を中心として面を閉じるように(展開時とは逆の方向に同じ角度だけ)回転させる。また、その辺が展開の際に回転の軸となった情報は消し、組み立て途中の図形からも再び展開のプロセスがとれるようにしておく。どの面を回転させるかは3章(4)を行うアルゴリズムと同様にDFSから求められるので、回転後の幾何学データ及びそれまで切れていた辺が繋がったことによる位相データのアップデートを行い、図形をコンピュータ上に表示する。立体の辺の中で、切れているもの、あるいは展開における回転の軸となった情報を持っているものがなければ、立体は完全に組み上がっていることになる。

5. おわりに

任意の立体図形を展開したり、組み立てたりするための基本データ構造及びアルゴリズムについて簡単に述べた。展開・組み立てにおいては面と面の接続関係が変化するので、その探索(DFS)とアップデートを効率良く行うことが重要である。

なお、本研究は製品化された立体幾何学習用ソフトウェア「IBM数学ラボシリーズ=3Dラボ」に応用された[2]。

【参考文献】

- [1] 長谷容子：“幾何学習支援ツールにおける図形展開機能”. CAI学会第18回全国大会講演論文集, pp. 199-202, 1993.
- [2] 日高一義：“立体図形学習のためのツール型ソフトウェアの開発”. 第48回情報処理学会全国大会, 1994.