

有限要素生成における粗分割処理への述語型言語の適用

1 R-4

中元勝也* 渡辺邦子** 松本茂年* 齊藤晃義** 和田孝*

*日立計測エンジニアリング(株) ** (株)日立製作所ソフトウェア工場

1. はじめに

有限要素法で使われる要素を形状モデルから生成する過程は、通常粗分割と密分割の二段階からなる。まず、粗分割では、有限要素に比べて、比較的大きな部分領域(以降、粗部分領域と呼ぶ)に分割する。粗部分領域は、後に続く密分割で、有限要素に分割される。

密分割では、補間を使って粗分割領域の境界の情報から内部の点を計算し、有限要素を生成する。そのためには、粗部分領域は、二次元の場合には三角形とか四角形、三次元の場合には四面体、五面体、六面体などの補間が簡単にできる形状でなければならない。密分割は、補間の計算が主体なので、ほぼ自動的に処理できる。

この理由から、形状モデルから有限要素を発生する過程で人手を多く消費するのは粗分割の部分であり、粗分割を容易にする要請が出てくる。

粗分割を容易にする一つの方法は、粗部分領域を自動的に見出すことである。これを粗部分領域の自動探索という。形状モデルは、頂点、稜線、面の情報を持っているので、これを使うことによって、六面体等の粗部分領域として許された形状を形状モデルからコンピュータだけで自動的に見出せる可能性がある。

筆者らは、形状モデルから粗部分領域を自動的に探索する処理をPrologで使われている述語論理を使って簡潔に実現し、カスタマイズしやすく信頼性の高い自動探索機能を実現する見通しを得たので、これについて報告する。

2. 実現方法

探索の処理は、一般に探索条件の一致を見ながら、不一致なら元に戻って再度一致を調べる処理が必要である。AIの分野でよく利用されているPrologは、一致(ユニフィケーション)と、元に戻る(バックトラック)機能を持ち、探索に必要な機能を備えている。筆者らは、この点に着目し、Prologが持っているユニフィケーションとバックトラック機能を備え、ホーン節を処理し、ゴール節に基づいた結果を得ることができる手続き(mesh/Prologと呼ぶ)を試作し、日立のCADシステムHICAD/Wに組み込み、テストした。粗分割だけを目的にしたので、一般のPrologが持っている標準的な関数は持っていない。

図1は、筆者らが使ったHICAD/Wの構成である。

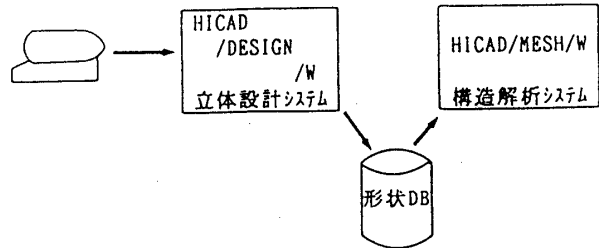


図1 HICAD/Wの構成

形状モデルはHICAD/DESIGN/Wで作成され、形状DBに登録される。HICAD/MESH/Wは、形状モデルを有限要素に分割するプログラムである。

HICAD/MESH/Wは、粗部分領域の稜線を、逐一指定して粗分割する機能を備えている。先に述べた自動探索の処理をこのHICAD/MESH/Wの粗部分領域を定義する部分に組み入れた。

図2は、組み込んだ処理の構成を示したものである。

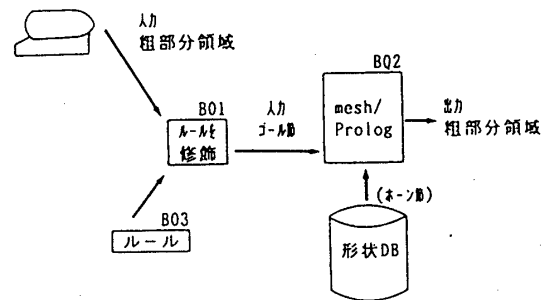


図2 自動探索処理の構成

B03のルールは、Prologのゴール節に対応するもので、例えば六面体を定義するルールは次の形をしている。

hex(E1,E2,E3,E4,E5,E6,E7,E8,E9,E10,E11,E12)
 :-evv(E1,V1,V2),evv(E2,V2,V3),evv(E3,V3,V4),
 ,evv(E4,V4,V1),evv(E5,V5,V1),evv(E6,V6,V2),
 ,evv(E7,V7,V3),evv(E8,V8,V4),evv(E9,V5,V6),
 ,evv(E10,V6,V7),evv(E11,V7,V8),evv(E12,V8,V
 5),noteq(E1,E2,E3,E4,E5,E6,E7,E8,E9,E10,E1
 1,E12). - (1)

ここで、 $evv(e_1, v_1, v_2)$ は、稜線 e_1 が頂点 v_1 と v_2 を結ぶことを意味する。 e_1 と v_1, v_2 のすべてが与えられている場合は、 v_1 と v_2 を結ぶ稜線が実際に存在すれば evv の値は真、そうでなければ偽となる。 e_1, v_1 又は v_2 のいずれか、又はすべてが未決定である場合は、 evv の値が真になるように未決定な部分を決定する。これが真になると、一つ右の evv へ進み、最後まで進むと六面体が定まることを (1) 式は示している。noteq はパラメタ中のすべての項が違っていることを調べている関数で、稜線が重複して選ばれないために必要である。また、一般的な Prolog の約束として、大文字と小文字はそれぞれ未知数と既知数に対応させている。

B03 のルールの evv のパラメタはすべて未決定の状態である。

図 2 の入力部分で粗部分領域の一部が指定されると、その情報でルールの一部を修飾して、未決定から既知 (与えられたもの) に変える。これが、B02 の入力になる。

例として、図 3 の場合を説明する。

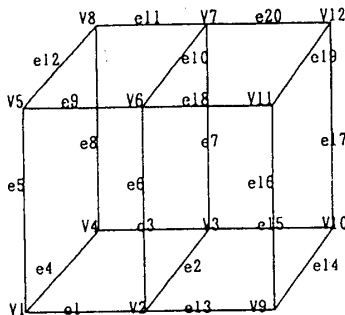


図 3 構造物の例

ユーザが e_{13} を指示した場合に、B02 に入力されるゴール節は次のようになる。

?-hex(e_{13} ,E2,E3,E4,E5,E6,E7,E8,E9,E10,E11,E12). - (2)

一般の Prolog では、ゴール節は、ゴール節の前に定義されたホーン節とユニファイしながら真偽を決定するが、筆者らのシステムでは、形状モデル DB のデータとユニファイする。B02 の中では、 e_{13} に結合した e_1, e_2, e_6 を E2 の候補として調べるが、E2 として使われるのは e_2 か e_6 のいずれかである。 e_1 を選んだ場合は、ゴール節の中の evv のいずれかが偽となり、六面体は完成しない。この場合は、 e_{13} だけが確定した状態に戻る。これをバックトラックといっている。こうすることによって、粗部分領域を見出すことができる。

全体は、C 言語で記述している。B02 の部分は、ホーン節とゴール節を入力して、その真偽を見出す一般的な仕様を持っており、有限要素を生成するための特定の形になっていないことが、筆者らが試作した手続きの特徴で、次の長所がある。

- (1) ルールの入替えが容易である。例で述べた六面体だけでなく、四面体や五面体など他の形状にも簡単に適用できる。これを使うとメーカーだけでなく、ユーザが独自のルールも入力できるので、カスタマイズしやすいシステムが実現できる。
- (2) 一般化した形のユニフィケーションとバックトラックを使っているので処理の論理的基盤が明確で、結果として信頼性が高い。

4. 実際例

具体的なデータを使って探索時間を調べた。使用したモデルの外形図を図 4 に示す。

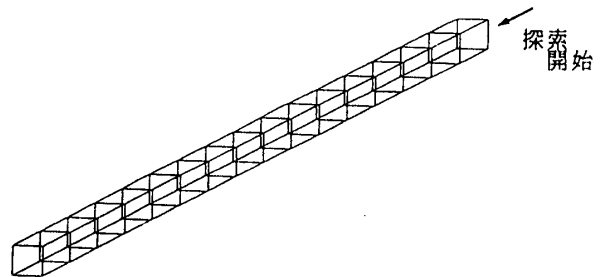


図 4 性能測定に使用したモデルの外形図

10MIPS のマシンで換算した場合、(1) 式の簡単なルールを使うと六面体一つの探索は最短で約 4.6 秒、最長で約 17.8 秒であった。

5. おわりに

Prolog に似たロジックを使うことによって、粗分割領域を自動探索する手続きを簡潔に実現できる見通しを得た。10MIPS の CPU を使うと 20 個程度の六面体の場合、約 4 分程度で探索できる。これは粗部分領域すべてを手手で定義するより約 4 倍速い。位相情報だけでなく、幾何情報も加えたルールを使うことによって、検索時間を更に短縮し、大きいデータに適用できるようにすることが今後の課題である。

6. 参考文献

- [1] Richard H.Gallagher 著、川井忠彦 監訳；「ギャラガー有限要素解析の基礎」、丸善、1976.
- [2] 人工知能 (AI) の進歩と適用へのアプローチ特集号、「オートメーション」、日刊工業新聞社、May、vol.30、No.5、1985.
- [3] インテリジェント CAD 特集号、「コンピュータール」、コロナ社、No.25、1989.