(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3624016号
(P3624016)
(45)発行日　平成17年2月23日(2005.2.23)　(24)登録日　平成16年12月3日(2004.12.3)

(51)Int.Cl.[7]	F I		
G06T 17/40	G06T	17/40	A
G01B 11/24	G06F	17/50	622C
G06F 17/50	G06F	17/50	624F
	G01B	11/24	K

請求項の数 18　(全 18 頁)

(21)出願番号　特願平7-153278
(22)出願日　平成7年6月20日(1995.6.20)
(65)公開番号　特開平9-6992
(43)公開日　平成9年1月10日(1997.1.10)
審査請求日　平成14年6月20日(2002.6.20)

(73)特許権者　000001007
キヤノン株式会社
東京都大田区下丸子３丁目３０番２号
(74)代理人　100076428
弁理士　大塚　康徳
(74)代理人　100093908
弁理士　松本　研一
(72)発明者　佐々木　豊成
東京都大田区下丸子３丁目３０番２号　キヤノン株式会社内

審査官　伊知地　和之

最終頁に続く

(54)【発明の名称】有限要素分割モデルから輪郭線、頂点を抽出する方法及びその装置

(57)【特許請求の範囲】
【請求項１】
メモリ上の有限要素分割モデル領域に記憶された３次元の有限要素分割モデルから、複数の要素に共有されないフリーフェイスを抽出して前記メモリ上のフリーフェイス記憶領域に記憶し、
前記フリーフェイス記憶領域に記憶されたフリーフェイスのデータを使用して輪郭線を抽出することを特徴とする有限要素分割モデルから輪郭線を抽出する方法。
【請求項２】
前記フリーフェイスの抽出では、１つの要素のすべての面について面を共有している他の要素が存在するかどうかのチェックを行うことで、該要素からフリーフェイスを抽出し、
前記輪郭線の抽出では、要素の１つの辺について、その辺を一辺に持つ２つの面がフリーフェイスとして抽出されかどうかをチェックし、２つの面が共にフリーフェイスとして抽出されていたならば該辺を輪郭線と判定することを特徴とする請求項１記載の有限要素分割モデルから輪郭線を抽出する方法。
【請求項３】
前記輪郭線の抽出では、１つの要素から抽出されたフリーフェイスの数が１つ以下の場合に、該要素の辺はすべて輪郭線ではないと判定することを特徴とする請求項２記載の有限要素分割モデルから輪郭線を抽出する方法。
【請求項４】
メモリ上の有限要素分割モデル領域に記憶された３次元の有限要素分割モデルから、複数

の要素に共有されないフリーフェイスを抽出して前記メモリ上のフリーフェイス記憶領域に記憶し、
前記フリーフェイス記憶領域に記憶されたフリーフェイスのデータを使用して輪郭線を抽出して前記メモリ上の輪郭線記憶領域に記憶し、
前記輪郭線記憶領域に記憶された輪郭線のデータより頂点を抽出することを特徴とする有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項５】
前記フリーフェイスの抽出では、１つの要素のすべての面について面を共有している他の要素が存在するかどうかのチェックを行うことで、該要素からフリーフェイスを抽出し、
前記輪郭線の抽出では、要素の１つの辺について、その辺を一辺に持つ２つの面がフリーフェイスとして抽出されたかどうかをチェックし、２つの面が共にフリーフェイスとして抽出されていたならば該辺を輪郭線と判定し、
前記頂点の抽出では、各要素の１つの節点について、その点を端点に持つ辺が輪郭線として抽出されたかをチェックし、該辺のすべてが輪郭線として抽出されていたならば該節点を頂点と判定することを特徴とする請求項４記載の有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項６】
前記輪郭線の抽出では、１つの要素から抽出されたフリーフェイスの数が１つ以下の場合に、該要素の辺はすべて輪郭線ではないと判定することを特徴とする請求項５記載の有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項７】
前記頂点の抽出では、１つの要素から抽出された輪郭線の数が２つ以下の場合に、該要素の節点はすべて頂点ではないと判定することを特徴とする請求項５記載の有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項８】
更に、抽出された輪郭線のうち両端の節点が同じ節点番号からなる重複する輪郭線を抽出し、輪郭線の重複を要素分割モデル中の要素の不整合の検査に用いることを特徴とする請求項１記載の有限要素分割モデルから輪郭線を抽出する方法。
【請求項９】
輪郭線の表示処理において、抽出した重複した輪郭線を重複していない輪郭線と区別できるように、線の表示態様を違えて表示することを特徴とする請求項８記載の有限要素分割モデルから輪郭線を抽出する方法。
【請求項１０】
更に、抽出された頂点を要素分割モデルのチェックに用いることを特徴とする請求項４記載の有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項１１】
更に、輪郭線、または複数の要素に共有されない辺であるフリーエッジを表示した図に頂点を同時に表示し、分割モデルをチェックすることを特徴とする請求項１０記載の有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項１２】
メモリ上の有限要素分割モデル領域に記憶された２次元の有限要素分割モデルから、１つの要素のすべての辺について、その辺を共有している他の要素が存在するかどうかのチェックを行なうことで、複数の要素に共有されない辺であるフリーエッジを抽出して前記メモリ上のフリーエッジ記憶領域に記憶し、
前記フリーエッジ記憶領域に記憶されたフリーエッジのデータを用いて、各要素の１つの節点について、その点を端点に持つ２つの辺がフリーエッジであるかどうかをチェックし、２つの辺が共にフリーエッジであるならば該節点を頂点と判定することで、頂点を得ることを特徴とする有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項１３】
前記頂点の抽出では、１つの要素から抽出されたフリーエッジの数が１つ以下の場合に、

該要素の節点はすべて頂点ではないと判定することを特徴とする請求項１２記載の有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項１４】
更に、得られた頂点を要素分割モデルのチェックに用いることを特徴とする請求項１２記載の有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項１５】
更に、輪郭線またはフリーエッジを表示した図に頂点を同時に表示し、分割モデルをチェックすることを特徴とする請求項１２記載の有限要素分割モデルから輪郭線、頂点を抽出する方法。
【請求項１６】
１つの要素のすべての面について、その面を共有している他の要素が存在するかどうかのチェックを行うことで、該要素から複数の要素に共有されないフリーフェイスを抽出するフリーフェイス抽出手段と、
１つの要素から抽出されたフリーフェイスの数が１つ以下の場合に、該要素の辺はすべて輪郭線ではないと判定する第１輪郭線判定手段と、
１つの要素から抽出されたフリーフェイスの数が２つ以上の場合に、要素の１つの辺について、その辺を一辺に持つ２つの面がフリーフェイスとして抽出されたかどうかをチェックし、２つの面が共にフリーフェイスとして抽出されていたならば該辺を輪郭線と判定する第２輪郭線判定手段とを備えることを特徴とする有限要素分割モデルから輪郭線を抽出する装置。
【請求項１７】
１つの要素のすべての面について、その面を共有している他の要素が存在するかどうかのチェックを行うことで、該要素から複数の要素に共有されないフリーフェイスを抽出するフリーフェイス抽出手段と、
１つの要素から抽出されたフリーフェイスの数が１つ以下の場合に、該要素の辺はすべて輪郭線ではないと判定する第１輪郭線判定手段と、
１つの要素から抽出されたフリーフェイスの数が２つ以上の場合に、該要素の１つの辺について、その辺を一辺に持つ２つの面がフリーフェイスであったかどうかをチェックし、２つの面が共にフリーフェイスであったならば該辺を輪郭線と判定する第２輪郭線判定手段と、
１つの要素から抽出された輪郭線の数が２つ以下の場合に、該要素の節点はすべて頂点ではないと判定する第１頂点判定手段と、
１つの要素から抽出された輪郭線の数が３つ以上の場合に、該要素の１つの節点について、その点を端点に持つ辺が輪郭線として抽出されたかをチェックし、該辺のすべてが輪郭線として抽出されていたならば該節点を頂点と判定する第２頂点判定手段とを備えることを特徴とする有限要素分割モデルから輪郭線、頂点を抽出する装置。
【請求項１８】
１つの要素のすべての辺について、その辺を共有している他の要素が存在するかどうかのチェックを行い、該要素の複数の要素に共有されない辺であるフリーエッジを抽出するフリーエッジを抽出手段と、
１つの要素から抽出されたフリーエッジの数が１つ以下の場合に、該要素の節点はすべて頂点ではないと判定する第１頂点判定手段と、
１つの要素から抽出されたフリーエッジの数が２つ以上の場合に、各要素の１つの節点について、その点を端点に持つ２つの辺がフリーエッジとして抽出されたかどうかをチェックし、２つの辺が共にフリーエッジとして抽出されていたならば該節点を頂点と判定する第２頂点判定手段とを備えることを特徴とする有限要素分割モデルから輪郭線、頂点を抽出する装置。
【発明の詳細な説明】
【０００１】
【産業上の利用分野】

本発明は、有限要素解析，境界要素解析等に使用する２次元または３次元の有限要素分割モデルの表示および分割モデルの持つ誤りの発見に利用でき、具体的には有限要素分割モデルの輪郭線および頂点（３次元モデルの場合とがった部分、２次元モデルの場合角の部分）の抽出を行う方法及びその装置に関するものである。

【０００２】

【従来の技術】

近年計算機の急速な発達と共に、有限差分法，有限要素法，境界要素法等の数値解析が頻繁に行われるようになった。これらの数値解析法では、一般に、解析モデルを多数の有限要素に分割した有限要素分割モデルが使用される（ここでは、有限差分法，有限要素法，境界要素法等に使用する分割モデルをまとめて有限要素分割モデルと呼ぶ）。

【０００３】

有限要素分割モデルの表示において、有限要素モデルの輪郭線を抽出して表示することが多々ある。ここで、輪郭線とは、３次元有限要素モデルの場合にその稜線を示し、細かくは要素の辺（要素を構成する面の境界線）から構成されるものである。このように輪郭線を抽出して表示するのは、有限要素すべてを表示した場合に表示に時間がかかることに起因するものである。ｎ個の要素からなる３次元モデルの場合、その輪郭線の数は単純に考えればｎの３乗根となる。このことは表示を行う時間を大幅に短縮できることを意味し、視点位置を決定するための試行錯誤の段階における表示等においては特に有効である。また、時系列の解析結果のアニメーション表示においては、短時間に何枚もの図を表示する必要があるが、輪郭線を用いることで短時間に行えるようにしたという例もある（特開平４－３４６１７８号）。

【０００４】

また、輪郭線は、要素分割モデルに含まれる要素の不整合、例えば、要素の抜け（分割モデル中が要素で完全に満たされていない）、浮き節点（要素の辺上に節点がある）、節点の重複（同一点に節点が２つ以上ある）等の発見にも使用される。これは、輪郭線を表示することによって行われ、具体的には要素分割モデルの内部に存在する輪郭線を発見することによってなされる。すなわち、本来輪郭線は分割モデルの稜線だけに表示されるものであり、それが内部にあるということはその部分に要素の不整合があることを意味するからである。

【０００５】

従来、輪郭線は、複数の要素に共有されない辺（フリーエッジ）によって代用されていた。具体的には、輪郭線は、全要素を構成するすべての辺から２つ以上の要素に共有される辺を取り除くことにより抽出されていた。

以下に、図８に示した流れ図に従って、従来の輪郭線（フリーエッジ）を抽出する方法を説明する。

【０００６】

まず、ステップＳ８０１でチェックの対象とする要素（Ｅｉ）を設定し、ステップＳ８０２でその要素の辺を１つ設定する。次に、ステップＳ８０３でその辺と同一の辺を持つＥｉ以外の要素が存在するかどうかをチェックする。ここで、同一の辺とは両端の節点番号が同じであるという意味である。次に、ステップＳ８０４にてフリーエッジであるかどうかの判定を行う。ステップＳ８０３のチェックで、もし同一の辺を持つ要素が存在すれば、本辺はフリーエッジではない。なければフリーエッジである。フリーエッジならば、ステップＳ８０５でメモリに登録し、フリーエッジでなければステップＳ８０６に進み次の処理に移る。

【０００７】

以上の処理を要素Ｅｉのすべての辺について行うため、ステップＳ８０６で全辺完了をチェックし、完了していない場合はステップＳ８０２に戻り次の辺設定に行く。これらの処理を、ステップＳ８０７のチェックでステップＳ８０１に戻り、すべての要素について行う。すべての要素についてチェックを行った後、ステップＳ８０８で、登録されたフリーエッジ（輪郭線）をグラフィックディスプレイに表示する。

【０００８】
本処理において、ステップＳ８０６からＳ８０２へのループの反復回数は、要素Ｅｉを構成する辺の数だけ行うことになる。図９に主要な３次元要素の辺の数を示す。
図８の流れ図中の処理で、ステップＳ８０３で示したフリーエッジの判定は、辺の両端の節点番号の和が同じ辺だけをチェックの対象とすることにより、モデルの要素数に依存しないでほぼ一定時間に処理できることが、特開平４－３４６１７８号に記載されている。
【０００９】
図８の流れ図から従来の方法では、フリーエッジの抽出に要する時間が次式に比例することがわかる。
【００１０】
【数１】

$$\sum_{i=1}^{elem} L_i$$

ここで、Ｌｉは要素Ｅｉの辺の数、ｅｌｅｍは要素数である。
【００１１】
次に、図１０の分割モデルを用いて、フリーエッジを抽出する過程を更に具体的に説明する。説明は図８に沿って行う。
図１０は４つの３角柱の要素からなる分割図であり、図中Ｅ１～Ｅ４は要素番号を示す。黒丸は節点を示し、節点近傍に記載した番号は節点番号である。要素Ｅ１～Ｅ４はそれぞれ以下の節点からなる。
Ｅ１：１－３－４－５－７－８
Ｅ２：１－２－３－５－６－７
Ｅ３：５－７－８－９－１１－１２
Ｅ４：５－６－７－９－１０－１１
本分割モデルから、以下の手順でフリーエッジを抽出する。
（１）要素Ｅ１を設定する（ステップＳ８０１）。
（２）要素Ｅ１の辺１－３を設定する（ステップＳ８０２）。
（３）辺１－３を持つ要素が他にあるかどうかをチェックする（ステップＳ８０３）。要素Ｅ２が同一の節点からなる辺を持っているので、辺１－３はフリーエッジではないと判定される（ステップＳ８０４）。
（４）辺１－３がフリーエッジではなかったので、ステップＳ８０６からＳ８０２に戻って、新たな辺３－４を設定する（ステップＳ８０２）。
（５）辺３－４について同様にフリーエッジのチェックを行う（ステップＳ８０３）。この辺と同一の辺を持つ要素は発見されず、本辺をフリーエッジと判定する（ステップＳ８０４）。
（６）辺３－４をフリーエッジとして登録する（ステップＳ８０５）。
（７）（２）～（６）の操作を、要素Ｅ１の残りすべての辺について繰り返し実行する。これによって、更に辺４－１，４－８がフリーエッジとして登録される。
（９）以上の操作を、残りの要素Ｅ２～Ｅ４についてステップＳ８０７からＳ８０１戻るループで行う。その結果、更に９つの辺（１－２，２－３，２－６，８－１２，６－１０，１１－１２，１２－９，９－１０，１０－１１）がフリーエッジとして登録される。
【００１２】
図１１は本操作によって抽出されたフリーエッジを描画したものである（ステップＳ８０８）。図中、図１０で隠線として破線で表示したものは破線のままで表示している。節点１－９，３－１１を結ぶ線についてはモデルの稜線であるにもかかわらず表示されていないが、これはフリーエッジの特徴（複数の要素に共有されない辺）であるため仕方がない。

【００１３】
本操作において、図８のステップＳ８０７からＳ８０１のループで示した反復は４回、ステップＳ８０６からＳ８０２のループの反復は９回となり、合計３６回の反復が行われる（式（１）において、Ｌｉはすべて９であり、要素数は４である）。
以上３次元モデルの場合について説明したが、本輪郭線は２次元の有限要素モデルについても同様に有効であり、表示の高速化、分割の不整合等の発見に使用することができる。２次元モデルの場合、輪郭線は解析モデル（解析領域）の境界線を示す。輪郭線は、３次元モデルの場合と同様にフリーエッジ（複数の要素に共有されない辺）と対応する。そして、３次元モデルの場合は、図１１で示したように輪郭線でありながらフリーエッジとして抽出されない辺があったが、２次元モデルの場合、輪郭線はフリーエッジと完全に一致する。２次元モデルにおけるフリーエッジは、３次元モデルと同様に図８の流れ図に示す手順で抽出することができる。
【００１４】
ここで、図１２に２次元要素分割モデルのフリーエッジを抽出した例を示す。（ａ）図は要素分割モデルを示す。Ｅ１～Ｅ４の４つの４辺形要素からなっており、各要素の構成節点は以下の通りである。
Ｅ１：１－８－６－２
Ｅ２：２－６－７－３
Ｅ３：４－７－９－５
Ｅ４：６－８－９－７
本モデルでは、本来節点３，４が同一点であるべきであり、データ作成ミスにより斜線で示した三角形領域３－４－７に要素の抜けが発生したものとする。本分割図のフリーエッジを表示したのが、（ｂ）図である。節点３－４－７を結ぶ輪郭線が抽出されており、これにより要素分割モデルの不整合を抽出できる。
【００１５】
なお、ここでは全要素を表示対象としたが、モデルによってはその一部の要素だけを表示対象としてもよい。すなわち、各要素が属性（材料特性等）をもっているような場合、一部の属性を持つ要素だけを表示対象として輪郭線を抽出してもよい（以後同様である）。
【００１６】
【発明が解決しようとする課題】
しかしながら、上記従来のフリーエッジを用いた輪郭線の抽出方法では、３次元モデルの場合に、解析モデルに含まれている要素分割の不整合を発見できない場合があった。このことを、図１３の要素分割モデルを用いて以下に示す。
図１３は４つの３角柱の要素からなる分割図であり、図中、Ｅ１～Ｅ４は要素番号を示す。黒丸及びその近傍に記載した数字は図１０におけるそれと同じである。要素Ｅ１～Ｅ４はそれぞれ以下の節点からなる三角柱の要素である。
Ｅ１：１－２－４－５－６－８
Ｅ２：２－３－４－６－７－８
Ｅ３：５－７－８－９－１１－１２
Ｅ４：５－６－７－９－１０－１１
本要素分割モデルは、節点５－６－７－８を含む面で要素分割が不整合となっている。すなわち、要素Ｅ１，Ｅ２については節点６－８を結ぶ線で要素分割がなされているのに対して、要素Ｅ３，Ｅ４では節点５－７を結ぶ線で要素分割がなされている。従って、面５－６－７－８において要素の面が不連続となっている。
【００１７】
本要素分割モデルに対して、従来の方法によりフリーエッジを抽出して表示したのが図１４である。１２本のフリーエッジが抽出・表示されている。ところが、図１４で表示されたフリーエッジはすべて輪郭線と呼べるものである（分割モデルの稜線である）。従って、本フリーエッジの図から分割モデル中の不整合を発見することはできない。一般に、従来のフリーエッジを用いた方法では、３次元モデルの場合に、上で述べた分割モデル中の

要素抜け，浮き節点，節点の重複については発見できたが、このような要素の面の不連続性については発見できなかった。

【００１８】

一方、２次元モデルの場合についても、要素の不整合を発見できない場合があった。例えば、図１２において要素Ｅ２が解析領域全体に対して非常に小さく、特に節点３，４，７が非常に近接している場合、（ｂ）図に示したフリーエッジの節点３，４，７のくぼみは非常に小さいものとなる。その結果、そのくぼみを見過ごしてしまうことがあった。

【００１９】

本発明は、このような課題に対処することを目的になされたものであり、３次元モデル中の要素の面の不連続性（２次元モデル中の微少なミスも含めて）、要素の不整合を容易に発見できる有限要素分割モデルから輪郭線、頂点を抽出する方法及びその装置を提供することを目的とする。

【００２０】

【課題を解決するための手段】

上記目的を達成するために、本発明の有限要素分割モデルから輪郭線を抽出する方法は、メモリ上の有限要素分割モデル領域に記憶された３次元の有限要素分割モデルから、複数の要素に共有されないフリーフェイスを抽出して前記メモリ上のフリーフェイス記憶領域に記憶し、前記フリーフェイス記憶領域に記憶されたフリーフェイスのデータを使用して輪郭線を抽出することを特徴とする。

【００２１】

又、本発明の有限要素分割モデルから輪郭線、頂点を抽出する方法は、メモリ上の有限要素分割モデル領域に記憶された３次元の有限要素分割モデルから、複数の要素に共有されないフリーフェイスを抽出して前記メモリ上のフリーフェイス記憶領域に記憶し、前記フリーフェイス記憶領域に記憶されたフリーフェイスのデータを使用して輪郭線を抽出して前記メモリ上の輪郭線記憶領域に記憶し、前記輪郭線記憶領域に記憶された輪郭線のデータより頂点を抽出することを特徴とする。

ここで、前記フリーフェイスの抽出では、１つの要素のすべての面について面を共有している他の要素が存在するかどうかのチェックを行うことで、該要素からフリーフェイスを抽出し、前記輪郭線の抽出では、要素の１つの辺について、その辺を一辺に持つ２つの面がフリーフェイスとして抽出されたかどうかをチェックし、２つの面が共にフリーフェイスとして抽出されていたならば該辺を輪郭線と判定し、前記頂点の抽出では、各要素の１つの節点について、その点を端点に持つ辺が輪郭線として抽出されたかをチェックし、該辺のすべてが輪郭線として抽出されていたならば該節点を頂点と判定する。

【００２２】

又、前記輪郭線の抽出では、１つの要素から抽出されたフリーフェイスの数が１つ以下の場合に、該要素の辺はすべて輪郭線ではないと判定する。又、前記頂点の抽出では、１つの要素から抽出された輪郭線の数が２つ以下の場合に、該要素の節点はすべて頂点ではないと判定する。

更に、抽出された輪郭線のうち両端の節点が同じ節点番号からなる重複する輪郭線を抽出し、輪郭線の重複を要素分割モデル中の要素の不整合の検査に用いることを特徴とする。更に、輪郭線またはフリーエッジを表示した図に頂点を同時に表示し、分割モデルをチェックする。

【００２３】

又、本発明の有限要素分割モデルから輪郭線、頂点を抽出する方法は、メモリ上の有限要素分割モデル領域に記憶された２次元の有限要素分割モデルから、１つの要素のすべての辺について、その辺を共有している他の要素が存在するかどうかのチェックを行なうことで、複数の要素に共有されない辺であるフリーエッジを抽出して前記メモリ上のフリーエッジ記憶領域に記憶し、前記フリーエッジ記憶領域に記憶されたフリーエッジのデータを用いて、各要素の１つの節点について、その点を端点に持つ２つの辺がフリーエッジであるかどうかをチェックし、２つの辺が共にフリーエッジであるならば該節点を頂点と判定

することで、頂点を得ることを特徴とする。
ここで、前記頂点の抽出では、１つの要素から抽出されたフリーエッジの数が１つ以下の場合に、該要素の節点はすべて頂点ではないと判定する。更に、輪郭線またはフリーエッジを表示した図に頂点を同時に表示し、分割モデルをチェックする。
【００２４】
又、本発明の有限要素分割モデルから輪郭線を抽出する装置は、１つの要素のすべての面について、その面を共有している他の要素が存在するかどうかのチェックを行うことで、該要素から複数の要素に共有されないフリーフェイスを抽出するフリーフェイス抽出手段と、１つの要素から抽出されたフリーフェイスの数が１つ以下の場合に、該要素の辺はすべて輪郭線ではないと判定する第１輪郭線判定手段と、１つの要素から抽出されたフリーフェイスの数が２つ以上の場合に、要素の１つの辺について、その辺を一辺に持つ２つの面がフリーフェイスとして抽出されたかどうかをチェックし、２つの面が共にフリーフェイスとして抽出されていたならば該辺を輪郭線と判定する第２輪郭線判定手段とを備えることを特徴とする。
【００２５】
又、本発明の有限要素分割モデルから輪郭線、頂点を抽出する装置は、 １つの要素のすべての面について、その面を共有している他の要素が存在するかどうかのチェックを行うことで、該要素から複数の要素に共有されないフリーフェイスを抽出するフリーフェイス抽出手段と、１つの要素から抽出されたフリーフェイスの数が１つ以下の場合に、該要素の辺はすべて輪郭線ではないと判定する第１輪郭線判定手段と、１つの要素から抽出されたフリーフェイスの数が２つ以上の場合に、該要素の１つの辺について、その辺を一辺に持つ２つの面がフリーフェイスであったかどうかをチェックし、２つの面が共にフリーフェイスであったならば該辺を輪郭線と判定する第２輪郭線判定手段と、１つの要素から抽出された輪郭線の数が２つ以下の場合に、該要素の節点はすべて頂点ではないと判定する第１頂点判定手段と、１つの要素から抽出された輪郭線の数が３つ以上の場合に、該要素の１つの節点について、その点を端点に持つ辺が輪郭線として抽出されたかをチェックし、該辺のすべてが輪郭線として抽出されていたならば該節点を頂点と判定する第２頂点判定手段とを備えることを特徴とする。
【００２６】
又、本発明の有限要素分割モデルから輪郭線、頂点を抽出する装置は、１つの要素のすべての辺について、その辺を共有している他の要素が存在するかどうかのチェックを行い、該要素の複数の要素に共有されない辺であるフリーエッジを抽出するフリーエッジを抽出手段と、１つの要素から抽出されたフリーエッジの数が１つ以下の場合に、該要素の節点はすべて頂点ではないと判定する第１頂点判定手段と、１つの要素から抽出されたフリーエッジの数が２つ以上の場合に、各要素の１つの節点について、その点を端点に持つ２つの辺がフリーエッジとして抽出されたかどうかをチェックし、２つの辺が共にフリーエッジとして抽出されていたならば該節点を頂点と判定する第２頂点判定手段とを備えることを特徴とする。
【００２７】
【作用】
かかる構成により、要素分割モデルの表面（フリーフェイス）の情報に着目し、１要素中で隣合う面がフリーフェイスであるとき、その２面の両面は必ず解析モデルの輪郭線となるべきであることから、要素の面が不連続な部分に対しても輪郭線を得るようにしたものである。また、本発明における頂点は元来輪郭線の角の部分に位置すべきものであり、それが他の位置にあることにより、その頂点の部分に要素の不整合があることがわかるようにしたものである。
【００２８】
【実施例】
以下、本発明の実施例を添付図面に従って詳細に説明する。
図７は本実施例でフリーエッジを抽出して表示するのに使用される計算機システムのブロ

ック図である。
図中、７１はグラフィックディスプレイ等の表示部であり、７２は演算・制御用の中央処理装置（ＣＰＵ）、７３は補助記憶用のＲＡＭであり、有限要素分割モデルのデータを記憶する有限要素分割モデル領域７１ａと、抽出されたフリーフェイスを記憶するフリーフェイス記憶領域７１ｂと、抽出されたフリーエッジ（輪郭線）を記憶する輪郭線記憶領域７１ｃと、抽出された重複する輪郭線を記憶する重複輪郭線記憶領域７１ｄと、抽出された頂点を記憶する頂点記憶領域７１ｅとを含む。
【００２９】
７４はキーボードやポインティングデバイス等のデータ入力や動作指示、あるいはフロッピディスクや通信等の有限要素分割モデルのデータの入力を行う入力部である。また、７５はＣＰＵ７２の制御手順を格納するＲＯＭである。通常、入力部７４からデータを入力し、ＣＰＵ７２はＲＯＭ７５に格納された処理手順に従って、ＲＡＭ７３を用いて処理を行い、結果を表示部７１に表示する。
【００３０】
（実施例１）
図１は本発明の第１の実施例を説明する輪郭線を抽出するための処理の流れを示すである。以下に、図１に従って輪郭線を抽出する過程を説明する。
まず、ステップＳ１０１でチェックの対象とする要素（Ｅｉ）を設定する。次に、ステップＳ１０２～Ｓ１０６により要素Ｅｉのフリーフェイスを抽出する。すなわち、ステップＳ１０２にて要素Ｅｉの１つの面を設定し、ステップＳ１０３でその面と同一の面を持つＥｉ以外の要素が存在するかどうかを調べる。ここで、同一の面とは面を構成する節点の番号が同じであるという意味である。次に、ステップＳ１０４にてフリーフェイスであるかどうかの判定を行う。ステップＳ１０３のチェックで、もし同一の面を持つ要素が存在すれば、本面はフリーフェイスではない。なければフリーフェイスである。フリーフェイスならばステップＳ１０５にて登録を行い、フリーフェイスでなければステップＳ１０６に進んで次の処理に移る。以上の処理を、要素Ｅｉのすべての面についてステップＳ１０６からＳ１０２に戻って行う。これによって、要素Ｅｉのすべての面についてフリーフェイスであるかどうかが解る。
【００３１】
次に、ステップＳ１０７で要素Ｅｉのフリーフェイス数が１つ以下か否かを判定し、１つ以下の面から輪郭線は構成されないことから、本要素の辺はすべて輪郭線ではないと判定して、ステップＳ１２２に進み、次の要素に移る。
２つ以上のフリーフェイスがあった場合は、ステップＳ１０８で要素Ｅｉの１つの辺（Ｌｊ）を設定する。また、ステップＳ１１９で、Ｌｊを１辺に持つ要素Ｅｉの２つの面（Ｌｊを稜線として持つＬｊの両側の２つの面）がフリーフェイスであるかどうかを調べ、２つの面がフリーフェイスであったならば、ステップＳ１２０でＬｊを輪郭線として登録する。そうでなければＬｊは輪郭線ではないので、ステップＳ１２１に進む。本チェックをステップＳ１２１からＳ１０８に戻って、要素Ｅｉのすべての辺について行う。
【００３２】
以上の操作を、ステップＳ１２２からＳ１０１に戻るループにより全要素について行った後、抽出された輪郭線をステップＳ１２３で表示する。
以上の処理の中で、ステップＳ１０３に示したフリーフェイスであるかどうかの判定は、面を構成する節点番号の和が同じ面だけをチェックの対象とすることで、モデルの要素数に依存しないでほぼ一定時間で処理できることが特開平４－３４６１７８号に記載されている。
【００３３】
本処理におけるステップＳ１０６からＳ１０２のループの反復回数は、要素Ｅｉを構成する面の数である。主要な３次元要素の面の数を図９に示す。また、図１のステップＳ１２１からＳ１０８のループの反復回数は、要素Ｅｉを構成する辺の数である。しかし、一般的な要素分割モデルでは、ほとんどの要素が分割モデルの内部に位置することから、フリ

ーフェイスとなる面を２つ以上持たない。その結果、大部分の要素はステップＳ１０７からＳ１２２に進み、本ループが実行されることは非常に少ない。

【００３４】

従って、本方法で輪郭線を抽出した場合の時間は、次式にほぼ比例する。

【００３５】

【数２】

$$\sum_{i=1}^{elem} F_i$$

ここで、Ｆｉは要素Ｅｉの面の数、ｅｌｅｍは要素数である。

【００３６】

図９から３次元要素では面の数は辺の数よりも常に小さいことがわかる。従って、（２）式は常に（１）式よりも小である。すなわち、本実施例によれば、従来のフリーエッジを抽出する方法よりも少ない時間で輪郭線を抽出できる。

また、ここで抽出した輪郭線は隣合うフリーフェイスの稜線として抽出したものであるため、要素の面が不連続になっている分割モデルの不整合も抽出することが可能である。

【００３７】

従来例の項で使用した図１３の分割モデルを用いて、本実施例による輪郭線の抽出過程およびこのことを具体的に説明する。

本例では以下の手順で輪郭線を抽出する。まず、フリーフェイスの登録を行う。

（１）要素Ｅ１を設定する（ステップＳ１０１）。

（２）要素Ｅ１の面１－２－４を設定する（ステップＳ１０２）。

（３）面１－２－４を持つ要素が他にあるかどうかをチェックする（ステップＳ１０３）。その結果、本面と同一の面を持つ要素は発見されず、本面をフリーフェイスと判定する（ステップＳ１０４）。

（４）本面１－２－４をフリーフェイスに登録する（ステップＳ１０５）。

（５）ステップＳ１０６からステップＳ１０２に戻り、新たな面２－４－８－６を設定する。

（６）面２－４－８－６について同様にフリーフェイスのチェックを行う（ステップＳ１０３）。要素Ｅ２が同一の節点からなる面を持っているので、面２－４－８－６はフリーフェイスではないと判定される（ステップＳ１０４）。

（７）面２－４－８－６がフリーフェイスではなかったので、ステップＳ１０６を経て再び新たな面を設定する。

（８）（２）～（７）の操作を要素Ｅ１の残りすべての面に実行する。これによって、更に３つの面（１－５－８－４，１－５－６－２，５－６－８）がフリーフェイスとして登録される。

【００３８】

次に、フリーエッジの登録に移る。

（９）要素Ｅ１に対して合計４面がフリーフェイスとして登録された。そこで、ステップＳ１０７を経て、要素Ｅ１の１辺１－４を設定する。（ステップＳ１０８）

（１０）辺１－４を稜線として持つ２面１－２－４と１－５－８－４とは共にフリーフェイスなので、ステップＳ１１９の判定を経て、辺１－４を輪郭線として登録する（ステップＳ１２０）。

（１１）ステップＳ１２１からＳ１０８に戻って、新たな辺２－４を設定する。

（１２）辺２－４を稜線として持つ２面１－２－４，２－４－８－６のうち後者はフリーフェイスではないので（ステップＳ１１９）、ステップＳ１２１を経て更に新たな辺の設定を行う（ステップＳ１０８）。

（１３）（１０）～（１２）の操作を要素Ｅ１の残りのすべての辺について行う。その結

果、新たに辺１－２，１－５，５－８，５－６が輪郭線として登録される。
（１４）以上の操作を、ステップＳ１２２からＳ１０１に戻って、残りの要素Ｅ２～Ｅ４について行う。その結果、更に１１本の辺（２－３，３－４，３－７，７－８，６－７，８－１２，９－１２，１１－１２，６－１０，９－１０，１０－１１）がフリーエッジとして登録される。なお、ここで辺５－６，６－７，７－８，８－５は２度登録される。

【００３９】

図２は本操作によって抽出した輪郭線を描画したものである（ステップＳ１２３）。図中、図１０で隠線として破線で表示したものは破線のままで表示している。図２には、明らかに輪郭線ではない線５－６，６－７，７－８，８－５がある。これから、節点５，６，７，８で囲まれた領域では要素の結合が連続でないことがわかる。

【００４０】

なお、本実施例では輪郭線の抽出をフリーフェイスの抽出と同時に行っている。すなわち、図１におけるステップＳ１０５で登録される面をすべて保存しておけば、すべてのフリーフェイスを抽出したことになり、フリーフェイスと輪郭線を同時に抽出できたことになる。フリーフェイスは解析モデルの表面を示すものであり、３次元要素分割モデルを隠線表示する場合等において非常に有用なものである。

【００４１】

また、従来、フリーフェイスは輪郭線とは別のプログラムで抽出していたものであるが、本方法によりフリーフェイスと輪郭線との抽出プログラムを１つにまとめることができたことになる。

（実施例２）

第１実施例では抽出した輪郭線を表示することにより要素の不整合を発見した。しかし、要素中の不整合の存在は、抽出した輪郭線に重複するものがあるかどうかをチェックすることで容易に発見可能である。次に、このことについて詳しく述べる。

【００４２】

実施例１では１つの要素中で隣接する２つのフリーフェイスの稜線を輪郭線として抽出した。従って、ここで抽出された輪郭線の両側の面は必ずフリーフェイスである。そして、輪郭線が重複するということはその辺（輪郭線）を持つ２つの要素の面が異なることを示す（同一ならばフリーフェイスにならない）。ここで、有限要素モデルの要素がすべて面でつながっている（１つの辺だけでつながっているようなくびれの部分はない）という条件を加えると、その２つの面は同一面上にあることを意味する。従って、その部分では必ず要素の面が不連続となっているといえる。

【００４３】

すなわち、実施例１で抽出した輪郭線はもし要素に不整合がない（要素と要素が１つの面で接合している）ならば必ずユニークである（重複するものはない）はずであり、重複する輪郭線がある部分では必ずその部分で要素間の面が不連続となっている。
そこで本実施例２は、実施例１で抽出した輪郭線の中から重複するものを抽出することで要素分割モデルの不整合をチェックし、実施例１における要素の不整合の発見を更に容易にする。

【００４４】

図３に重複する輪郭線を抽出するための処理の流れを示す。尚、本処理は、図１に追加されて各要素単位で行われても、図１の全要素のチェック後に行われても良い。
まず、ステップＳ３０１で輪郭線の１本（Ｏｉ）を設定し、ステップＳ３０２でその輪郭線と重複する（両端の節点番号が同じである）輪郭線の有無をチェックする。そして、輪郭線Ｏｉと重複するものがあれば本輪郭線をステップＳ３０４で登録し、なければステップＳ３０５に進み次の処理に移る。ステップＳ３０５からＳ３０１に戻るループで、本処理をすべての輪郭線に対して行うことにより、すべての重複した輪郭線を抽出する。

【００４５】

ここで、ステップＳ３０２の重複チェックは基本的にはＯｉを除くすべての輪郭線に対して行わなくてはならないが、あらかじめ両端の節点番号の和が同一の輪郭線をグループ化

しておき、同一の和を持つ輪郭線だけをチェックの対象とすることで時間短縮をはかることが可能である。
以上の処理で抽出した重複する輪郭線を、輪郭線を表示する過程（図１のステップＳ１２３）で判別できるように表示する。判別できるように表示することで、その部分で要素に不整合が存在していることが容易にわかるようになる。
【００４６】
なお、重複している輪郭線と重複していない輪郭線とを判別できるようにする方法としては、両者間で表示する線の色，線種（破線，鎖線，波線，２重線等），線の太さのうち少なくとも１つを変化させて表示するとよい。線の色を変化させることで、異常（要素の不整合）を知らせる警告効果が高まる。また線種，線の太さを変化させる方法では、白黒のグラフィックスディスプレイにも対応が可能となる。
【００４７】
また重複している輪郭線だけを表示してもよい。重複した輪郭線を表示することにより、モデル中の要素の不整合の存在を確認することができる。
また輪郭線の表示を行わなくても、重複した輪郭線の有無をチェックすることで、モデル中に含まれる要素の不整合の有無だけは調べることができる。このことは、計算を行う前の入力データのチェックを行うのに有効である。
【００４８】
（実施例３）
第１の実施例では要素のフリーフェイスから輪郭線を抽出した。このことを更に進めれば、本輪郭線から分割モデルの頂点（モデルのとがった部分）を抽出することができる。そして、頂点を用いることで要素分割モデルの要素の不整合を更に確実に発見できるようになる。このことを図２，図４を用いて説明する。尚、図４の処理は、図１に追加されて各要素単位で行われても、図１の全要素のチェック後に行われても良い。
【００４９】
ここでは、要素Ｅｉのすべての辺について輪郭線であるかどうかのチェックを行った後、輪郭線が２本以下の場合には、２本以下の辺で頂点が形成されることはないことから、ステップＳ４０１にて、要素Ｅｉの構成節点はすべて頂点ではないと判定し、処理は行わない。一方、輪郭線が３本以上抽出された場合は、ステップＳ４０２にてＥｉの構成節点の１つ（Ｎｉ）を設定する。そして、ステップＳ４０３で、節点Ｎｉを端点として持つ要素Ｅｉの辺がすべて輪郭線であったかどうかを判定し、すべてが輪郭線であったならば、ステップＳ４０４で節点Ｎｉを頂点として登録する。輪郭線でない辺があったならば、ステップＳ４０５の処理に移る。ステップＳ４０５からＳ４０２に戻って、本操作を要素Ｅｉのすべての構成節点について行う。
【００５０】
本操作を図１に組み込んで、すべての要素について、フリーフェイス，輪郭線の抽出と同時に行うことにより、分割モデルのすべての頂点が抽出される。
以上の操作を、図１３の分割モデルを用いて具体的に説明する。
（１）まず、要素Ｅ１の輪郭線の数をチェックし（ステップＳＳ４０１）、５本であることから節点１を設定する（ステップＳ４０２）。
（２）節点１を端点に持つ３つの辺１－５，１－４，１－２が輪郭線であるかどうかをチェックする（ステップＳ４０３）。
（３）３つの辺が共に輪郭線であることから、節点１を頂点として登録する（ステップＳ４０４）。
（４）次に、ステップＳ４０２に戻って、節点２を設定し、節点２を端点に持つ３つの辺１－２，２－４，２－６が輪郭線であるかどうかをチェックする（ステップＳ４０３）。
（５）辺２－４（及び辺２－６）が輪郭線でないことから、ステップＳ４０５に進んでステップＳ４０２に戻り、次の節点を設定する。
（６）以上の処理を、要素Ｅ１のすべての節点について行う。その結果、節点１，５が頂点として抽出される。

（７）本操作を、残りの要素Ｅ２～Ｅ４についてもフリーフェイス，輪郭線の抽出処理の後に行う。

【００５１】

以上の処理の結果、節点１，３，５，６，７，８，１０，１２の８つの点が抽出される。図２に本操作により抽出された頂点を節点を示す黒丸を丸で囲んで示した。ここで抽出された頂点は、要素分割モデルに不整合等がなければ分割モデルのとがった部分に位置するものであり、また複数の要素に共有されることはない。

【００５２】

ところが本例の場合、頂点５，６，７，８が分割モデルの稜線の上にあり、このことから容易にこの部分で要素分割が不整合になっていることがわかる。実施例１では、輪郭線のみを表示することで分割モデル中の要素の不整合を見つけたが、このように頂点も同時に表示することで、更に確実に不整合を発見できる。

（実施例４）

実施例１，２，３を用いて、本実施例の３次元分割モデルの場合の一例について、フリーフェイスを介して、輪郭線，頂点と順に次元の低い要素を構成する素を抽出する方法について説明してきた。このことを２次元要素分割モデルに応用すれば、２次元モデルからフリーエッジを介して容易に頂点を抽出することができる。本実施例４では、２次元要素分割モデルから頂点を抽出して分割モデルのチェックを行う方法について説明する。

【００５３】

本実施例による２次元分割モデルから頂点を抽出する過程を図５に示す。図５は、図１における面を辺に、辺を節点に置き換えたものに等しい。従って、ここではその処理過程の詳細は省略し、概要だけを説明する。

まず、図５のステップＳ５０１にて要素（Ｅｉ）を設定し、ステップＳ５０２～Ｓ５０６の処理にてその要素Ｅｉの辺からフリーエッジを抽出する。そして、フリーエッジの抽出結果を用いて、ステップＳ５０７～Ｓ５１１にて頂点を抽出する。ステップＳ５０７では要素Ｅｉのフリーエッジの数が１つ以下ならば、１つ以下の辺で頂点は形成されないことから、頂点はないと判定してステップＳ５０７からＳ５１２に進む。また、ステップＳ５０９の判定では節点を端点に持つ２つの辺が共にフリーエッジであったかどうかをチェックする。

【００５４】

このようにして求めた頂点は、３次元モデルの場合と同様、もし要素分割モデルに不整合がなければ、分割モデル（解析領域）の角の部分に位置するものであり、複数の要素に共有されることはない。そこで、本頂点を輪郭線に同時に表示し、その位置をチェックすることにより、分割モデル中の要素の不整合を発見することができる。

【００５５】

図６は、図１２の（ａ）に示した要素分割モデルから、本方法を用いて抽出した頂点をフリーエッジと共に表示した図である。頂点を丸で囲んで示した。

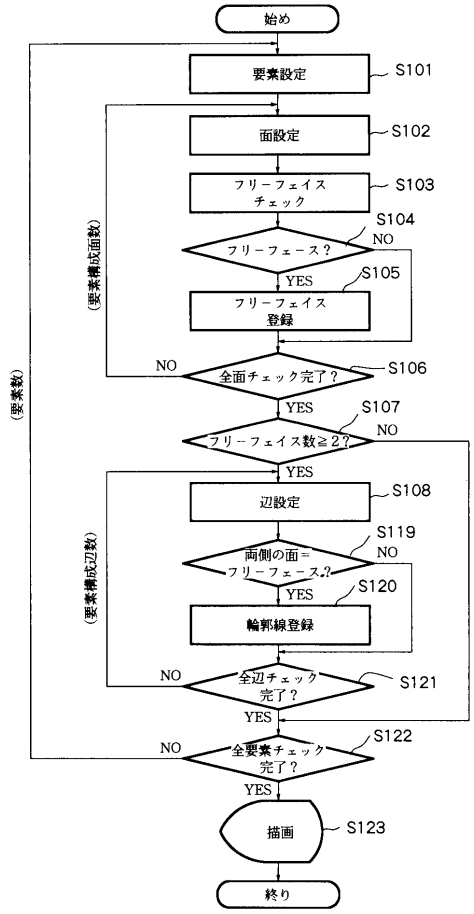
本図では辺１－５の間に頂点３，４が位置している。このことから、頂点３，４の部分に要素の不整合（この場合はモデルの角）があることがわかり、容易にデータミスがあることが発見できる。なおこのことは、図１２の（ａ）において要素Ｅ２が解析領域全体に対して非常に小さく、図１２の（ｂ）における節点３，４，７のフリーエッジのくぼみが非常に小さい場合にも、必ず頂点３，４が抽出されることから有効である。

【００５６】

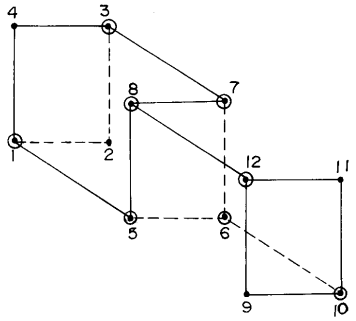
尚、上記本実施例は互いに組み合わせることにより、更に効果的な有限要素分割モデルの不整合の発見が可能となる。また、実施例においては、有限差分法，有限要素法及び境界要素法に使用される要素について説明したが、本発明はこれらと同様な構造を有するデータに対して適用しうることは明らかである。本発明は、複数の機器から構成されるシステムに適用しても、１つの機器から成る装置に適用しても良い。また、本発明はシステム或は装置にプログラムを供給することによって達成される場合にも適用できることは言うまでもない。

【００５７】
【発明の効果】
以上の結果から明らかなように、本発明により、３次元モデル中の要素の面の不連続性（２次元モデル中の微少なミスも含めて）、要素の不整合を容易に発見できる有限要素分割モデルから輪郭線、頂点を抽出する方法及びその装置を提供できる。
【００５８】
詳細には、３次元モデルについて本発明による輪郭線の抽出法、及び抽出した輪郭線を用いることで、以下のような効果が生まれた。
（１）従来の輪郭線（フリーエッジ）では発見できなかった要素分割モデルの不整合を発見できるようになった。
（２）従来よりも短時間で輪郭線を抽出できるようになった。
（３）輪郭線を抽出する段階で、要素分割図の表示等に有用なフリーフェイスも同時に抽出できた。
（４）フリーフェイスと輪郭線を抽出するプログラムを１つにまとめることができた。その結果、プログラムの管理を簡略化できた。
【００５９】
また、本発明で抽出した輪郭線の重複を調べることで、以下の効果が生まれた。
（１）単なる輪郭線の表示よりも更に容易に要素分割の不整合を発見できるようになった。
（２）輪郭線を表示させることなく、不整合の有無がわかるようになった。
【００６０】
また、本発明で抽出した頂点を輪郭線と同時に表示することで、更に確実に要素の不整合をチェックできるようになった。
【図面の簡単な説明】
【図１】本発明の実施例１の処理の流れ示す図である。
【図２】本発明の実施例１の処理結果を示す図である。
【図３】本発明の実施例２の処理の流れを示す図である。
【図４】本発明の実施例３の処理の流れを示す図である。
【図５】本発明の実施例４の処理の流れを示す図である。
【図６】本発明の実施例４の適用例を示す図である。
【図７】本実施例の計算機の構成を示すブロック図である。
【図８】従来の処理の流れを説明する図である。
【図９】反復回数を説明する図である。
【図１０】実際の有限要素分割モデルにおける例を説明する図である。
【図１１】実際の有限要素分割モデルにおける例を説明する図である。
【図１２】実際の有限要素分割モデルにおける例を説明する図である。
【図１３】実際の有限要素分割モデルにおける例を説明する図である。
【図１４】実際の有限要素分割モデルにおける例を説明する図である。
【符号の説明】
７１　表示部
７２　ＣＰＵ
７３　ＲＡＭ
７４　入力部
７５　ＲＯＭ

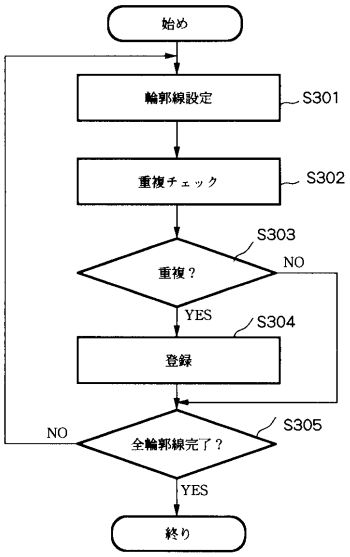
【図１】


始め
要素設定
S101
面設定
S102
フリーフェイス
チェック
S103
S104
フリーフェース？
NO
YES
S105
フリーフェイス
登録
(要素構成面数)
NO
全面チェック完了？
S106
YES
S107
フリーフェイス数≧２？
NO
YES
(要素数)
辺設定
S108
S119
両側の面＝
フリーフェース？
NO
YES
S120
輪郭線登録
(要素構成辺数)
NO
全辺チェック
完了？
S121
YES
NO
全要素チェック
完了？
S122
YES
描画
S123
終り

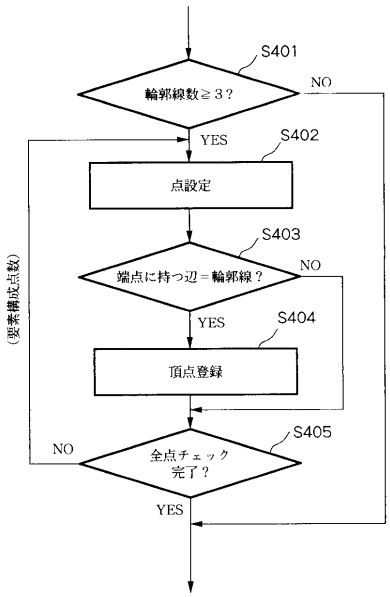
【図２】


4
3
8
7
1
2
12
11
5
6
9
10

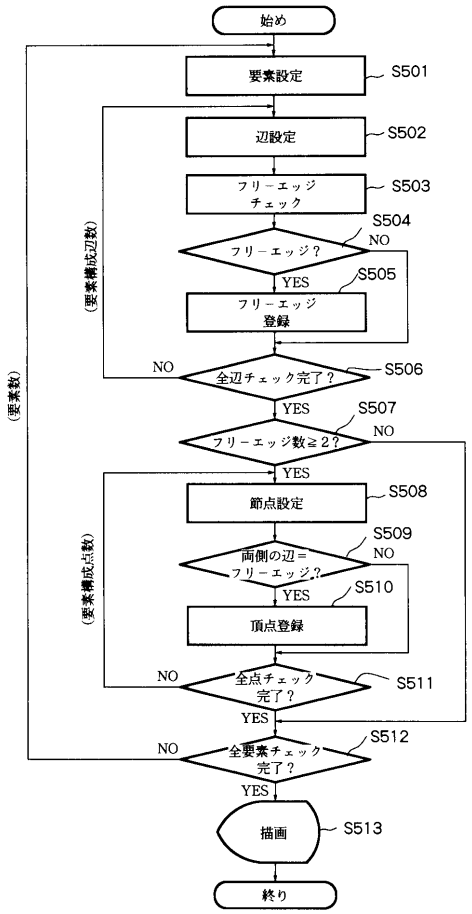
【図３】


始め
輪郭線設定
S301
重複チェック
S302
S303
重複？
NO
YES
S304
登録
NO
全輪郭線完了？
S305
YES
終り

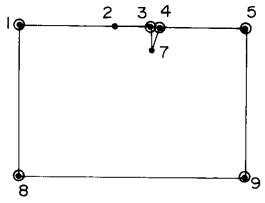
【図４】


S401
輪郭線数≧３？
NO
YES
S402
点設定
S403
端点に持つ辺＝輪郭線？
NO
YES
S404
頂点登録
(要素構成点数)
NO
全点チェック
完了？
S405
YES

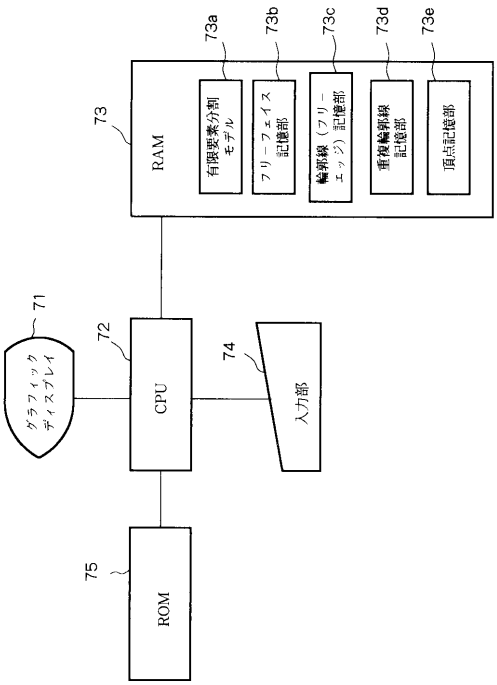
【図５】


始め
要素設定
S501
辺設定
S502
フリーエッジ
チェック
S503
S504
フリーエッジ？
NO
YES
S505
フリーエッジ
登録
全辺チェック完了？
S506
NO
YES
S507
フリーエッジ数≧2？
NO
YES
節点設定
S508
S509
両側の辺＝
フリーエッジ？
NO
YES
S510
頂点登録
全点チェック
完了？
S511
NO
YES
全要素チェック
完了？
S512
NO
YES
描画
S513
終り
（要素構成辺数）
（要素構成点数）
（要素数）

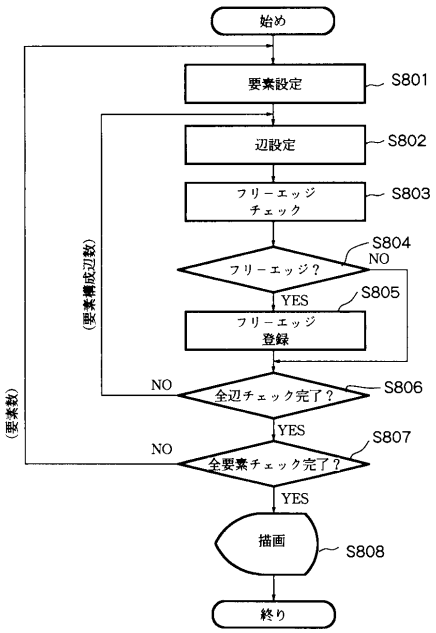
【図６】


1
2
3
4
5
7
8
9

【図７】


73a
73b
73c
73d
73e
73
RAM
有限要素分割
モデル
フリーフェイス
記憶部
輪郭線（フリー
エッジ）記憶部
重複輪郭線
記憶部
頂点記憶部
71
グラフィック
ディスプレイ
72
CPU
74
入力部
75
ROM

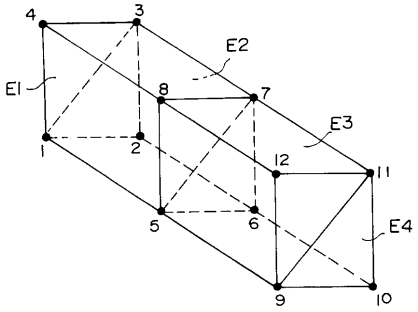
【図８】


始め
要素設定
S801
辺設定
S802
フリーエッジ
チェック
S803
S804
フリーエッジ？
NO
YES
S805
フリーエッジ
登録
全辺チェック完了？
S806
NO
YES
S807
全要素チェック完了？
NO
YES
描画
S808
終り
（要素構成辺数）
（要素数）

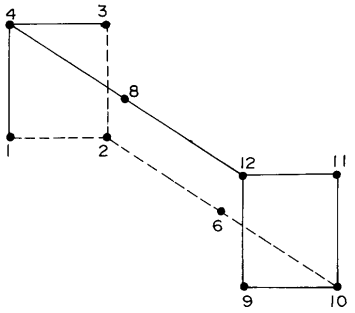
【図９】

要素形状	辺の数	面の数
３角錐	6	4
４角錐	8	5
３角錐	9	5
６角錐	12	6

【図１０】

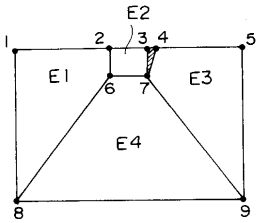


【図１１】

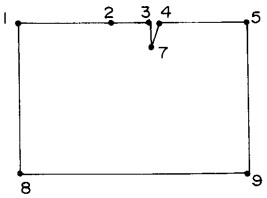


【図１２】

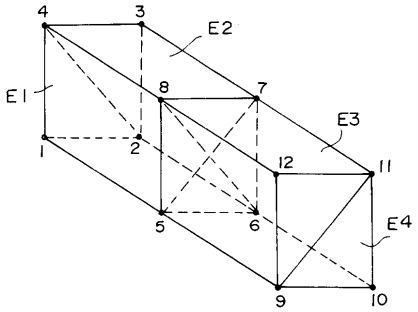
(a)



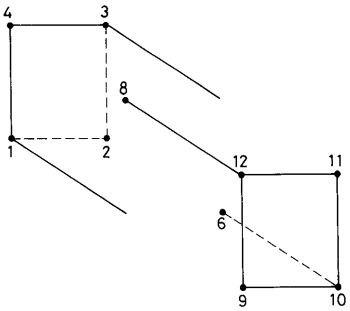
(b)



【図１３】



【図１４】



フロントページの続き

(56)参考文献 特開平０７－０８５３１０（ＪＰ，Ａ）
特開平５－２５０４４５（ＪＰ，Ａ）
特開昭６３－１６３８９（ＪＰ，Ａ）
特開平２－７３４７１（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
G06T 17/40
G06F 17/50
G06T 17/20
CSDB(日本国特許庁)