

CADにおける3次元構造物の隠線処理の一手法について

3S-8

西村直樹 竹内一博 *芝原輝夫 (富士通株式会社)
河野俊弘 鏡山英二 (㈱富士通九州システムエンジニアリング)

1. はじめに

3次元形状の図形が、実際には隠れて見えない線まで表示されていると、形状が一意的に定まらない場合が生じる。また、複雑な図形になると形状の把握が困難になることも少なくない。従って、3次元形状を扱うシステムでは、見えない稜線および面を消去して表現する機能が必要となる。

隠線処理の基本的な考え方は、物体のどの稜線、どの面が可視であるかを判定するという単純なものである。しかし、これをインプリメントするためには多大な演算時間を要する。この問題に関しては、これまでも数多くの研究がなされ、様々な方式が提案されている。本稿では、我々が考案した隠線処理の効率化のための工夫点について述べる。

2. 開発背景

(1) 従来からの基本的方法

隠線処理問題に対する基本的な方法としては、次の2つがある。

第一は、物体を多角形の集合体と考えて、図形出力装置上の各分解点で、どの稜線あるいは面が可視かを判定する方法である。この場合、各分解点において、すべてのn面についてどれが視点に最も近いかを判断する必要があり、処理回数は $n \cdot N$ に比例する。ただしNは図形出力装置の分解点の個数である。

第二の方法は、可視でない面あるいは面の一部を消去するために、n面と他の $n-1$ 面とを順次比較するものであり、その処理回数は $n \cdot (n-1)$ に比例する。実際、nおよびNの値は100000以上であり、さらに各ステップの処理にも相当の時間を要するため、演算時間は多大なものとなり、実用性に乏しい。従って、各ステップの処理を出来るだけ効率よくするように工夫することは重要なことである。

(2) 開発の要件

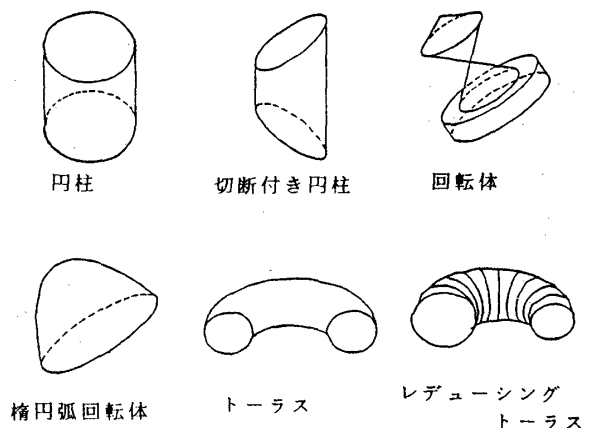
従来から開発されているものは、機械設計向けの本格的なソリッド・モデルをねらったものが大半である。そのため、使用メモリ・演算負荷は多大になり、取り扱える物体の個数は大きく制限されてしまう。

我々が対象としているプラント配管設計・機器配置設計においては、一度に取り扱う物体の個数が数千個以上と極めて多いことが特徴である。ここで要求されることは、実用に耐え得るように高速に隠線処理することである。さらには、出力結果がそのまま図面として使用出来るだけの精度が保障されていることも条件である。

3. 3次元形状の定義

3次元空間への物体の定義は、図1に示すような9種類のプリミティブを組合せてセグメントを作成することにより行う。各プリミティブのもつ情報は、稜線の線種・線幅・色等の図形属性情報および空間内の配置位置・配置方向・定義座標等の図形データ情報であり、従来のワイヤフレーム・モデルを拡張した形式のものである。

図1 プリミティブの種類

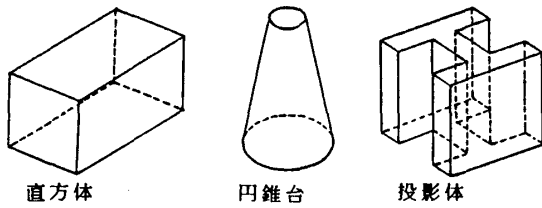


An Algorithm for Hidden-Line Elimination in 3D-CAD

Naoki Nishimura, Kazuhiro Takeuchi, Teruo Shibahara (*1)

Toshihiro Kawano, Eiji Kagamiyama (*2)

(*1) FUJITSU Ltd. (*2) FUJITSU KYUSYU SYSTEM ENGINEERING Ltd.



4. 隠線処理効率化の工夫

我々の採用した隠線処理の方法は基本的には、3次元の物体をすべて面に展開して、それぞれの面について他の面と比較して可視か可視でないかを判定するもので、2の(1)で述べた第二の方法に類似したものである。これを効率的に行うためには、隠線処理の対象となる面の組合せをあらかじめ減らしておくことが重要である。以下に処理対象削減の具体的方法を説明する。

(1) WINDOW分割

空間全体を一度に処理するのではなく、投影する平面上で適当な大きさの矩形領域（これをWINDOWと呼ぶ）に分割し、各々の領域内で面の比較を行うようにする。例えば、空間内に5000個のプリミティブが均等に分布しているとすれば、10個のWINDOWに分割して処理した場合、処理回数は10分の1になる。分割数を増やすことでさらに処理回数の削減が期待出来る。

(2) EXTENTチェック

面の比較を行う前に物体の空間的な拡がり（これをEXTENTと呼ぶ）を求めて、それらと比較する。空間的に全く重ならないのであれば、その組合せは除外する。EXTENTとして物体を完全に含む直方体を探るのが最も簡単であるが、隠線処理対象から除外出来るものをより多く検出するために、次のような特殊なEXTENTを考案した。

空間に定義された物体を任意の方向から見た場合のEXTENTとして図2のような2次元的なものを設定する。ここで、EXTENT定義座標系の座標軸 X_e 、 Y_e 、 Z_e は物体を定義する際の座標軸 X 、 Y 、 Z に各々垂直なものである。EXTENTが交わる条件は、 X_e 、 Y_e 、 Z_e の各々で交わっていることである。

(3) 輪郭線を利用した処理

次に交点計算を行う上で、処理回数を少なくするために空間に定義された各プリミティブの稜線を次の3つに分類する。

- (a) 模様線
可視面どうしの共通辺
- (b) 輪郭線
可視面と不可視面の境界をなす辺
- (c) 後方稜線
不可視面どうしの共通辺

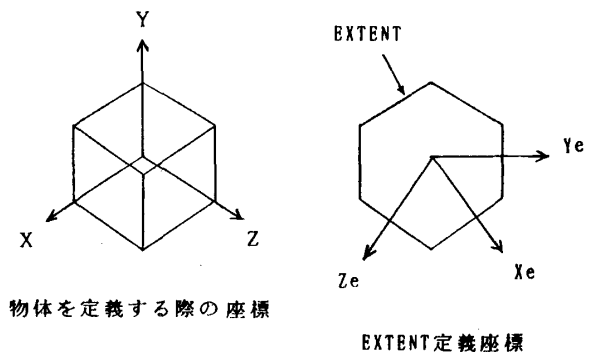
模様線は、他の輪郭線によって隠されることはあっても他を隠すことはない線である。輪郭線は、他の輪郭線および模様線を隠す可能性がある。後方稜線は、見えない稜線なので除外できる。従って、輪郭線と輪郭線・輪郭線と模様線の組合せの交点計算を行って交点情報を格納する。

これらの輪郭線データ・模様線データ・交点データを用いて隠線処理し、2次元の線分データを作成する。

5. 効果

上記以外にも細部にわたって処理に工夫を加えた結果、3000プリミティブからなるプラント配管図面を隠線処理するのにCPU時間3分（FACOM-M380を使用）で行えるようになり、実運用に十分耐え得るシステムを構築することが出来た。

図2 EXTENTの定義



物体を定義する際の座標

EXTENT定義座標