Computational Geometry for Structural Engineering

Etude de formes structurales à l'aide de l'ordinateur

Computerunterstützte geometrische Tragwerksmodellierung

Shizuo SHIMADA
Professor, Dr. of Eng.
Nagoya University
Nagoya, Japan

Shizuo Shimada, born in 1931, received his Dr. Eng. at the University of Tokyo. After five years as a Lecturer there he joined the Nagoya University in 1963 ans was promoted to full Professor in 1977. He has worked with bridge engineers as a specialist numerical analyst on many practical problems. Since 1983 he is managing the Division of Informatics which was founded in 1980.

SUMMARY

Mathematical studies on engineering shapes are reviewed in order to develop modeling procedures suitable for the computer. Geometry is discussed at three different levels, geometrical elements, figures and images. Dyadic product is introduced for the computation of moments of inertia and rotation matrices. The problem-oriented language GEOMETRY is proposed for educational and practical use in structural engineering.

RÉSUMÉ

L'étude mathématique de formes structurales est présentée en vue d'une utilisation raisonable de l'ordinateur. Les problèmes de géométrie sont traités à trois niveaux: éléments géométriques, figures et images. Le produit dyadique est utilisé pour le calcul à l'ordinateur du moment d'inertie et de la matrice de rotation. Le langage GEOMETRY est proposé pour l'enseignement et l'application pratique aux structures de génie civil.

ZUSAMMENFASSUNG

Ergebnisse mathematischer Untersuchungen über technische Formen werden zusammengestellt mit dem Ziel, den Computer zu deren Modellierung effektiver anzuwenden. Geometrie wird anhand von geometrischen Elementen, Figuren und Bildern diskutiert. Dyadische Produkte für die Berechnung von Trägheitsmomenten und Rotations-Matrizen werden eingeführt. Die problemorientierte Sprache GEOMETRIE wird für Ausbildung und praktische Anwendungen im konstruktiven Bauwesen vorgeschlagen.

1. INTRODUCTION

Geometry is a mathematical study on the properties of, and relations between, points, lines, surfaces and solids in space. Everybody is aware of some facts of elementary geometry as if they were self-evident truth. A word 'triangle' for instance is enough to remind us what properties it has. Even a child can compare two triangles at a glance whether similar or not. The mathematical conditions of similarity are rather troublesome and tricky, but applied in every field with or without remembering geometry.

Geometry becomes difficult without the aid of some analogical sketches. Since our living world is in three dimensional(3D) space, we recognize a spatial object through our eyes stereoscopically. We remember its characteristics so that we can identify it among others even though it appears in different ways. Two dimensional(2D) sketches are useful tools to record a shape and to remind it realistically. Normally, 2D sketches are insufficient media to tell the 3D information because of lacking in depth dimension. However, a human brain is so clever as to supply an associated information.

As far as a structural design is concerned, an engineer always imagines a structural shape that does'nt exist yet. He feels some difficulties to tell to another person what shape he is thinking about. He is obliged to prepare some written documents so that his ideas can be correctly transferred to those people who collaborate with him. If he works alone for his hobby, he may need no such documents and can directly tackle to materials to create a shape. A group of specialists with common knowledge among them use a few documents which are hardly understandable to other people. Provided a collaborator is a computer, it should be well taught to help us on the creative design works.

A computer suits for numerical processing, but not for topological solutions such as to find out some similarities among various shapes. A computer should therefore aid us on the computational problems in geometry. Subjects of the computational goemotry concern with the analyses of relative relations among points, lines and sufaces, the calculation of length, area or volume, the determination of tangential lines or surfaces, the generation of figures under some conditions, and so forth.

The computational geometry has been specially developed in the network and graph analyses with respect to the data bases of geography. A problem of Voronoi diagram(Thiessen graph or Dirichlet diagram) is one of interesting topics. Its principle of computation is very simple and clear. However, a lot of data require a reasonable algorithm for a practical use of a computer. This is the same situation in FEM analysis because it deals with a network structure. As for a structural engineering, classic problems on areas or volumes of structural members are of daily necessesity.

The author proposes in this paper a language GEOMETRY as a tool for the computational geometry. One of the aims of the language is at an education to young students as well as a practical use for engineers. Let us suppose again a triangle. We know well how to decide its centre of gravity, centre of inscribed circle and centre of circumscribed circle by the graphical means. However, we will feel a great difficulty to numerically decide these centres without the aid of sinusoidal functions. Ever to obtain a value of $\sin(x)$ or $\cos(x)$ was one of computational problems, but it has become so easy as being processed by a pocket calculator.

A little difficulty still remains to obtain the coordinate values of those centres. The language GEOMETRY can support such solutions. Moreover, it can provide an easy presentation on a graphic device just like a sketch on a paper. In order to obtain a reasonable solution about the analytical geometry, some standardized ideas are required for the data structure of points, lines, surfaces, solids and related mathematical values.

The author classifies his discussions about geometry into three steps. The first step deals with mathematical tools. "Geometrical elements" are their materials. A vector and a matrix are numerical tools to establish the methods of calculation among the geometrical elements. The second step defines a "figure". The author uses this word as the same sense as a shape. But strictly speaking, a figure is a geometrical model to represent a shape. A figure has a data set to draw an "image" on a sheet. The third step deals with a transformation from an invisible "figure" to a visible "image". The language GEOMETRY is introduced in the last chapter with some interresting topics. It is designed to meet for the author's discussions.

2 GEOMETRICAL ELEMENTS

2.1 Coordinate System

Point, line and plane are analytical elements of space geometry. These are directly given prior to any discussion. We know some of their characteristics such as a flat plane or a straight line. However, we can not define any units from which the elements are composed. Let us suppose two points, for instance, "A" and "B". There exists no numerical relation between two points whtether "A" is greater or less than "B".

For numerical handling in a computer, the elements shall be given by mathematical values so that some calculation, for instance, the intersect between two planes causing a line, can be performed. A Cartesian coordinate system is then supposed a priori in the world, that is, the world coordinate system(WC). This is also composed of three planes being perpendicular to each other. Intersected three lines are axes. The origin is a common point to all above elements. The unit length should be decided here. From now on, numerical characteristics of the elements are able to decide as relative relations against WC (Fig-1).

Two dimensional(2D) geometry is important in usual drawn images on a sheet. Point and line are analytical elements in 2D. The world becomes two dimensional, but it is recommended to suppose the third coordinate axis which is perpendicular to a sheet and gives a distance between your eyes and a sheet. When a sheet is laid on a desk, we consider x and y axes on a sheet. Sometimes, y and z axes more suit on a vertical CRT screen in connection with the space geometry.

Fig.1 Coordinate System

One dimensional(1D) geometry manages points along a line. This is not emphasized but a very important technique whenever we measure a distance between two points. An angle is measured by a curve linear 1D-coordinate system along a circle. Left/right or up/down is an associated property depending on the sign of a 1D-coordinate value.
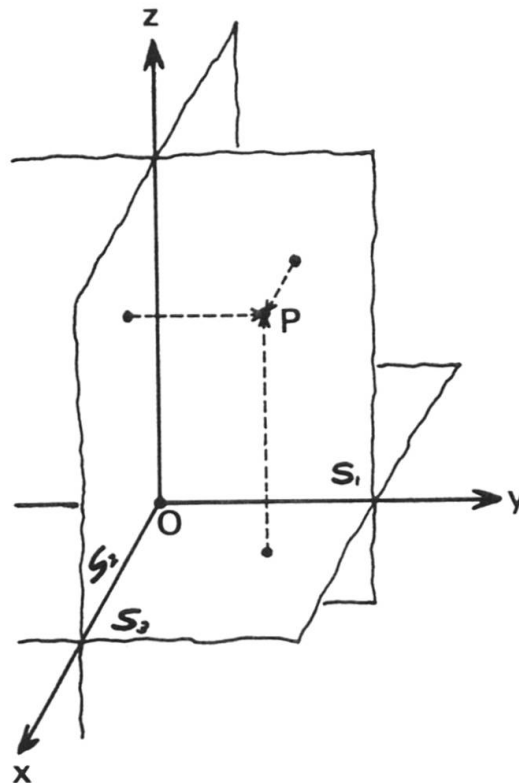
2.2 Point

A point is defined by a set of coordinate values in WC. These values present respectively a distance with sign between the given point and each coordinate plane. A set of values are then considered as a mathematical quantity called a a position vector from the origin. A vector in either 2D or 3D is to be managed as an array with size of 2 or 3 in the linear memories of a computer. A set of n-points are then extended in an array such as P(2,n) or P(3,n).

Since a computer has no spatial container to mark a point, an appropriate design is necessary for the data structure of the geometrical elements. A user needs little professional knowledge about programming technique. However, he should be careful not to refer the above arrays in an inverse style of P(n,2) or P(n,3). Geometry becomes a very difficult study when it is written by a symbolic expression. This is due to many such rules as to define an order or arrangement among values.
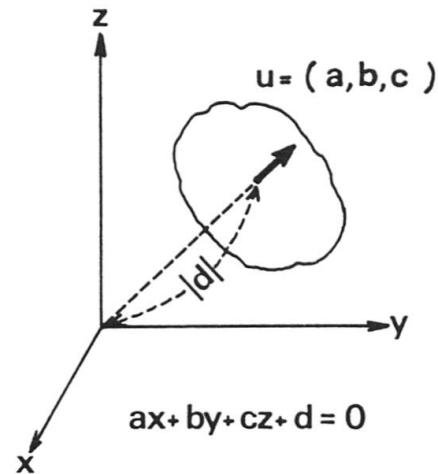

Fig.2 Definition of a plane

2.3 Plane

A flat plane extends infinitely dividing the world into two partial spaces relatively positive and negative. A plane has two faces, one of which appears practically as a surface of a solid body. An analytical tool of a plane is based on the following equation;

$$a x + b y + c z + d = 0 \qquad (1)$$

where, (x,y,z) is the coordinate values of a point in a plane. This can be rewritten into Eq.(2) using a product of two vectors.

$$(a,b,c).(x,y,z) + d = 0 \qquad (2)$$

Speaking geometrically, a vector (a,b,c) is perpendicular to the defined plane. Further, we decide this vector having unit length(Fig.2). Namely, the vecor is a normal vector to a plane. "d" becomes a distance with sign between the origin and a plane. The sign of d shows whether the origin stays in the positive or negative space relatively to the subjected plane. A set of four values (a,b,c,d) are then defined to represent a 3D-plane for numerical handling.

2.4 Line (3D)

A line is infinitely straight in the world. It must have a point(p,q,r) which is located at the closest against the origin. Starting at this point, another unit vector (u,v,w) is so proposed as to decide the direction of the given line. A set of six values (p,q,r,u,v,w) are then defined to represent a 3D-line. Equations for this line are(Fig-3);

$$x = p + u.t \ , \qquad y = q + v.t \quad \text{and} \quad z = r + w.t \qquad (3)$$

where, t is a local coordinate value along this line supposing 1D-geometry. Two vectors (p,q,r) and (u,v,w) are perpendicular each other in the world.
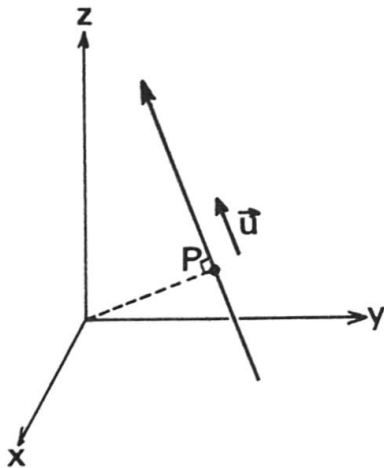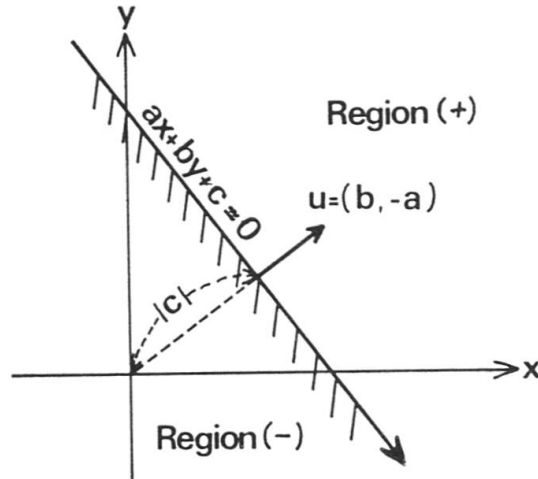
Fig.3 Defintion of a 3D-line   Fig.4 Defintion of a 2D-line

2.5 Line (2D)

A line drawn on a sheet has two properties with respect to a 3D-plane and a 3D-line, respectively. A line caused by the intersection between the sheet and a 3D-line divides the sheet into two regions. Equations 1 and 2 are reduced to those for 2D-geometry.

$$a.x + b.y + c = 0 \tag{4}$$

$$(a,b).(x,y) + c = 0 \tag{5}$$

The vector (a,b) defined in 2D-geometry has a unit size and is normal to the given line. "c" is a distance with sign between the 2D-origin and a 2D-line. A set of three values (a,b,c) are the definition for a 2D-line (Fig.4). Another way to difine a 2D-line is to project a 3D-line onto a sheet.

$$x = p + u.t \quad \text{and} \quad y = q + v.t \tag{6}$$

where (p,q) is the closest point along the line to the origin. The vector(u,v) has a unit length and indicates a positive direction of this line. This vector is perpendicular to the position vector(p,q). Equations 4, 5 and 6 present the same 2D-line.

3. INTERRELATION AMONG ELEMENTS

3.1 Angle and Distance

Even though the geometrical elements are defined in the previous chapter by a set of values, there exist no algebrailcal formulas among them on addition, subtraction, multiplication or division. There are topological relations such as left/right, inside/outside, inclusive/exclusive, coincident, common etc. In order to measure these topological properties, an introduction of angle and distance helps us more analytically and numerically rather than only to say left or right. It should be remembered again that the definitions of point, line and plane are relatively evaluated against the elements which construct the world coordinate system.

As to a distance, a vector plays an important role for measuring its magnitude. A unit vector has a unit length. A scalor is an algebraical multiplier to the unit vector. A topological propertiy, left or right, is then considered with respect to the sign of a scalor.

As to an angle, it measures a rotation around an axis. An axis is a 3D-line even when we consider a problem in 2D-geometry. A vector is necessary in order to represent an axis. A right-handed law is used to measure the magnitude and sign of an angle. A right screw is normally supposed to understand the law. A screw advances by its clockwise turn. A screw resembles a unit vector for an axis. An angle is a positive scalor when a screw advances. In a case of 2D-geometry, an axis stands perpendicularly to a sheet aiming at our side. It traces a point on a sheet. An angle is measured positive for a reversal clockwise turn.

3.2 Topological Status among Elements

Tables-1/2 summarlize the classified relations among geometrical elements in 3D and 2D, respectively. The status codes in Tables are also drawn in Figs.5a/5d. The interference between two geometrical elements is able to say by words such as coincident, exclusive, inclusive, etc. However, the computational geometry has to classify the status more strictly by numerical codes, because a word "parallel', for instance, has two or more different conditions.

From a viewpoint of numerical handling, a tolerance should be previously decided against a distance and an angle, respectively. For instance, a distance between a point and a plane is compared with a tolerance. If the distance is smaller than the tolerance, the point is in the plane practically. A tolerance for an angle is similarly used to check the parallelism among planes and lines. The condition has however a critical limit within a finite space. When a graphical presentation is concerned, such a tolerance is practically decided from the accuracy of a graphical screen.

Table 1. Classified relations among 3D-elements

element A vs. B	topological status	status code	measure	associated elements	remarks
Pt Pt	coincident	0			
	exclusive	1	dis	Ln	
Pt Ln	inclusive	0			
	exclusive	1	dis	Pt	closest point on a line
Pt Pl	inclusive	0			(Fig.5a)
	exclusive	1/2	dis	Pt	closest point on a plane
Ln Ln	coincident	0/1			(Fig.5c)
	parallel	2/3	dis	Pl	
	intersect	4	ang	Pt,Pl	co-plane
	exclusive	5	ang,dis	Pt,Pt	skew position
Ln Pl	inclusive	0			(Fig.5a)
	parallel	1/2	dis		
	intersect	3/4	ang	Pt	(Fig.5b)
Pl Pl	coincident	0/1			(Fig.5d)
	parallel	2/5	dis		
	intersect	6/7	ang	Ln	

* Pt(point), Ln(line), Pl(plane).

Table 2. Classified relations among 2D-elements

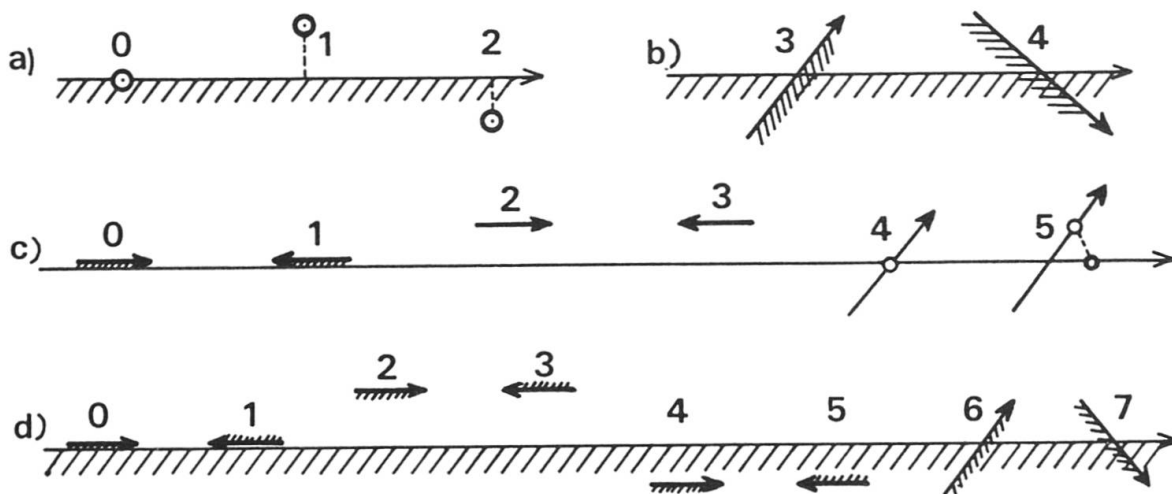| element A vs. B	topological status	status code	measure	associated elements	remarks
Pt Pt	coincident exclusive	0 1	 dis	 Ln	
Pt Ln	inclusive exclusive	0 1/2	 dis	 Pt	(Fig.5a) closest point on a line
Ln Ln	coincident parallel intersect	0/1 2/5 6/7	 dis ang	 Pt	(Fig.5d)

* Pt(point), Ln(line).

Fig.5 Status codes of topological relations

3.3 Mathematics of 3D-Vectors

A vector is a convenient mathematical quantity which is graphically modeled by an arrow showing its magnitude and direction. After deciding the world coordinate system, a vector enables us to calculate numerical properties of the geometrical elements. Among the formulas of vector mathematics, the author would like to explain some rules related to a product between two vectors. As to 2D-vector, the third component is zero or unused, practically.

In order to measure a relative angle between two vectors, a vector product and a scalor product are commonly used. The former is a special geometrical definition that produces a third vector perpendicular to the first and second vectors. A word 'product' normally means a scalor product because it is rather algebraical.

A dyad ab defined below is a sort of product between two vectors a and b. This product is represented by a matrix for numerical purposes.

$$ab = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} (b_1 \ b_2 \ b_3) = \begin{pmatrix} a_1 \cdot b_1 & a_1 \cdot b_2 & a_1 \cdot b_3 \\ a_2 \cdot b_1 & a_2 \cdot b_2 & a_2 \cdot b_3 \\ a_3 \cdot b_1 & a_3 \cdot b_2 & a_3 \cdot b_3 \end{pmatrix} \tag{7}$$

A dyad is used as an operator to produce another vector from one vector. Rules of product are the same as those for matrix algebra. A matrix is usually known as a tool for remembering a set of 3x3 components in row and column directions. A matrix is composed, in our discussion, of a series of dyads;

$$T = ab + cd + ef .... \qquad (8)$$

A rotaiton matrix is frequently used to rotate a position around an axis. It is mathematically carried out by a product between a matrix T and a position vector r.

$$r' = T.r \qquad (9)$$

Let us then show how T is constructed by a series of dyads. Let u be a unit vector representing a rotation axis. Any two unit vectors V and W are supposed so as to compose a Cartesian coordinate system (u,v,w). A position P which is given by a vector r is synthesized by the component vectors (Fig.6);

$$r = u(u.r) + v(v.r) + w(w.r)$$
$$= (uu + vv + ww).r \qquad (10)$$

After the rotation around a vector u, two vectors V and W change their direction in a plane which is normal to u;

$$v' = \quad v \cos(t) + w \sin(t)$$
$$w' = - v \sin(t) + w \cos(t) \quad , \quad (11)$$

Fig.6 Rotation of a point around an axis

A new position of P becomes then;

$$r' = u(u.r) + v'(v.r) + w'(w.r)$$
$$= (uu + (vv+ww)\cos(t) \qquad + (vw-wv))\sin(t)).r$$
$$= (E \cos(t) + uu(1-\cos(t)) + (vw-wv))\sin(t)).r \qquad (12)$$

Since two unit vectors V and W are temporary chosen, the componets of matrix T are calculated by the components of an axial vector (a,b,c) and an angle of rotation t;

$$T = \begin{pmatrix} 1 & & \\ & 1 & \\ & & 1 \end{pmatrix} \cos(t) + \begin{pmatrix} a.a & a.b & a.c \\ b.a & b.b & b.c \\ c.a & c.b & c.c \end{pmatrix} (1 - \cos(t)) + \begin{pmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{pmatrix} \sin(t) \qquad (13)$$

Equation 13 is practically useful to evaluate a rotation matrix T from a given axial vector u and an angle t of rotation. The above matrix T is used in numerical processing. In the meantime, a vector, which has a magnitude t and is proportional to u, is often considered to represent a rotation. A bending moment is drawn by such an axial vector for its similar property. An angle and a distance, introduced in the article 3.1, are therefore able to be considered as both vectors which are often drawn by arrow marks in a perspective sketch. However both vectors have each different property for numerical processing.
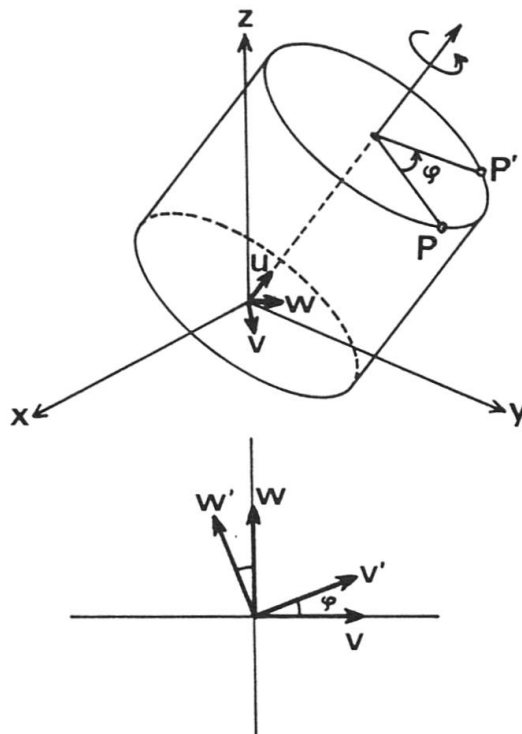
3.4 Moving and Rotating

Any two geometrical elements are able to superimpose one to another by moving and/or rotating. The topological status shown in Tables 1/2 defines the static relations between two elements. Moving and rotating are, on the other hand, dynamic procedures to cause such relations between the same element before and after its motion.

A word 'move' is used here in a sense of translation without 'rotate'. As to 'rotate', it needs a 3D-line for an axis which includes the origin of a coordinate system. Generally, an axis will not always meet with the origin. In such a case, 'rotate' is accompanied by 'move'.

A rotation of a position is carried out by a product between a rotation matrix T and a position vector r. A movement of a position is carried out by addition of a directional vector d. We shall define in general a motion of a point in two following steps 'rotate then move';

$$r' = T.r + d \tag{14}$$

where $r'=(x',y',z')$, $r=(x,y,z)$, $d=(dx,dy,dz)$ and T is a 3x3 matrix defined by Eq-13. For a numerical handling, a 3x4 matrix M is defined as a transformation matrix which has the vector d as the fourth column vector along with T. A matrix calculation is defined as;

$$(x',y',z') = M . (x,y,z,1) \tag{15}$$

Let us then consider the reversal procedure of Eq.14;

$$r = T' (r'-d) \tag{16}$$

This equation means that the procedure is carried out by 'move then rotate'. It is mathematically easy to transform this equation into 'rotate then move' expression. We should be careful to an order of procedure whether 'rotate then move' or 'move then rotate'.

3.5 Local Coordinate System(LC)

It is a natural manner to inform the size of a rectangle by its width and height. An implicit coordinate system is considered in this manner at the left bottom corner to measure its size. A local coordinate system is such a system, but should be strictly defined for the computational geometry.

A local coordinate system is a copy of WC. Its characteristics are summarized by a set of four vectors; e_1, e_2, e_3 and d. The first three vectors are unit vectors indicating the directions of new coordinate axes, respectively (Fig-7). The fourth vector d is the position of new origin. A 3x4 matrix M is used as a container of four vectors for numerical purposes.

$$M = \begin{pmatrix} e_{1x} & e_{1y} & e_{1z} & d_x \\ e_{2x} & e_{2y} & e_{2z} & d_y \\ e_{3x} & e_{3y} & e_{3z} & d_z \end{pmatrix} \tag{17}$$

A care should be paid to the arrangement of matrix components. The unit vectors are staying along row directions, while d stands in the fourth column. Although a rule to keep the 3x4 matrix components in computer memories uses a two dimensional array M(3,4), the components stay linearly as a series of four column vectors.

Using the definition of above matrix M, a position is transformed into a new position by Eq.15. The procedure has the follwing geometrical meaning. A local coordinate system was at the origin and equal to WC. The system has moved to a new position as defined by Eq.17. A subjected point to this local coordinate system changes its position in the world coordinate system. In other words, Eq.15 computes world coordinate values from locally defined coordinate values.

A reversal procedure is frequently required with respect to a local coordinate system. The problem is to compute the local coordinate values from the world coordinate values. In order to understand this procedure, let us suppose a TV-camera and a view through its lens. Looking at a TV screen in our room, we often feel an illusion that the view is in front of us realistically. The camera is, however, at another position and aiming at some direction. The definition of camera positioning is that of a local coordinate system. The camera measures a world position by its coordinate system. This is known as the coordinate transformation, geometrically (Fig.7).

Let us then suppose a camera aiming at an object in the world. A view through a lens only depends on relative positioning between the camera and the object. To take a fine picture, we move the camera to and fro. But the same result is possible by moving the object while the camera remains unmoved. Direction of motion is strictly opposite with respect to the camera or the object in WC.

Positioning of a camera in the world coordinate system is summarized by a word 'window' in a graphic language. After taking a film, a picture is moved in a local coordinate system so called a 'viewport'. A transformation matrix is used as a 2x3 matrix in a graphic language for a 2D domain. But it manages a scaling at the same time.
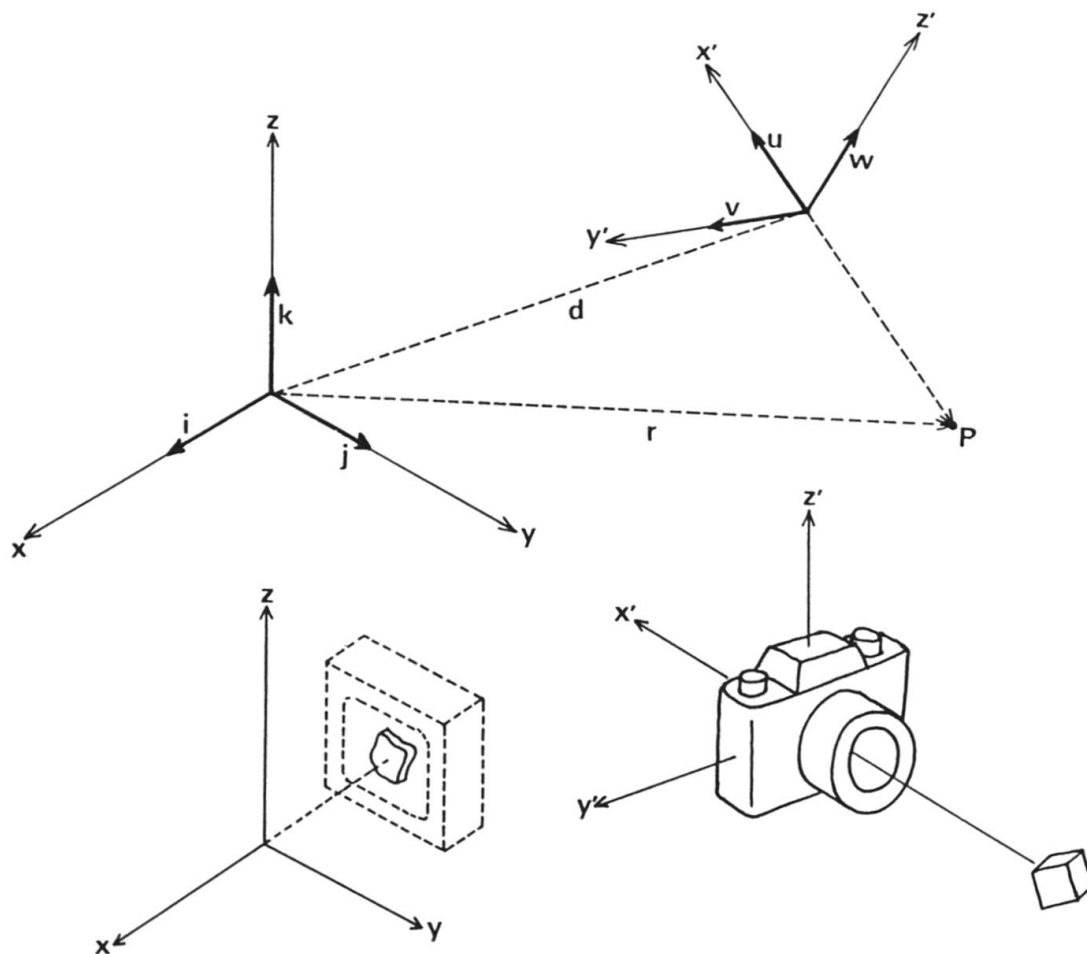

Fig.7 A local coordinate system

4. FIGURES --- MODELING OF SHAPES

4.1 Locus

Let us suppose a triangle. We know well that it is consisted of three edges. There are, however, many possibilities on presentation techniques on a sheet. Drawing an outline is a natural way. On the other hand, painting with a colour will emphasize its fill area and the outline may not be drawn. Sometimes, vertices are drawn by special symbols. Someone draws a triangle to show a hole. For such possibilities, we shall use two words 'figure' and 'image' with each separate geometrical definition.

A figure is represented by a word 'triangle' in the above paragraph. It has geometrical characteristics without the aid of any visible 'images'. A term locus suits to define what a figure is. A locus is a system of points, lines or surfaces representing a given condition or law. A circle, for instance, is a smooth curved line composed of a number of equi-distant 2D-points from a centre point. A word 'circle' is commonly used as a round outline as well as a shape whose outline is round. A clear difference should be recongnized between two circles. One is an outline and the other includes all the points inside of its outline.

An idea is necessary to classify a 2D-figure that it is composed of three figure elements; vertex, edge and region. A vertex is a point. An edge represents a line segment either straight or curved. A region is a fill area surrounded by edges. A triangle is consisted of three vertices, three edges, and a region if inside area is necessary.

The figure elements have common points among them. A vertex is common with two connecting edges. An edge between two vertices has an infinite number of points which are common with a region. Some definitions are necessary to identify whether some figure elements are vacant or not. If a region is vacant, a figure means an outline. If an outline is vacant, the points along edges are out of bounds of a figure. An outline is always necessary to decide a shape, therefore edges and vertices need the information on vacancy.

The concept of a locus is useful to understand what an image is. An 'image' is a visualized locus on a sheet or a graphic screen, because it is essentially a system of dotted points in black or colour. Drawing a point and a line segment corresponds to the visualization of vertex and edge, respectively. Painting on a region, however, takes a little special technique depending on a graphic device. Vacant figure elements are not drawn or may be drawn by dotted or dashed lines. A 2D-figure is able to transfer for the data to create an image, but an image needs more information such as thickness of line, colour, dotted lines, dashed lines, pattern species to fill area and so forth.

A solid shape is recognized by a 3D-figure. It is consisted of four 3D-figure elements; vertex, edge, surface and body which represents continuous fill materials. If 'body' is vacant, a thin-walled structure is assumed. A 3D-figure is also a locus, but it has topologically the same structure as that of a 2D-figure. Because the surfaces are bounded by each other through edges and can be developed like a map of the globe.

A 3D-figure becomes visible after a projective transformation into a 2D-figure and displaying graphically as an image. When a solid figure with a homogeneous fill material is transformed point by point into a 2D-figure, it will cause a silhouette without surface characteristics. Not all the points are transformed into a 2D-figure. Basically, fill materials inside a solid figure are out of our consideration. A special technique is required to remove invisible edges and surfaces and to make a realistic image. Hidden line elimination is discussed in article 5.4.

4.2 Modeling

Modeling procedures take the following three steps. First step is to establish an analogical model for the convenience of theoretical and numerical researches. The second is to decide its geometrical conditions. The last is to prepare its data for a computer. As far as the structural engineering is concerned, an FEM (finite element method) model is one of our targets.

An engineering structure is constructed by a lot of continuous materials. A finite number of points are chosen so that they represent mean properties around them. It is natural to consider a network connecting such points. The points are used as each origin of local coordinate systems by which the shapes of local materials are described (step-1).

A series of code is named to the points as well as to the edges. Coordinate values are set to the points under the assumption of a world coordinate system. Several attributive data are required such as point loading, forces, sectional areas, bending stiffness, and stress and deformation that shall be computed. Structural members have each local coordinate system to indicate their axes (step-2).

A design for a usage of computer memories is a modeling in narrow sense. Since the computer memories are in 1D-space, a reasonable idea is necessary to manage geometrical 2D or 3D data. The modeling is a method to describe a shape. A network model of FEM is simple but it has a purpose to compute stress and deformation. A precise description of a shape has practical requirement to make engineering drawings, to estimate amount of materials, and so on.

Several geometrical models are shown in Fig.8. A data structure for a 2D-map is similarly used to record such a 3D-solid as a polyhedron. A network model(2D) and a wire frame model(3D) become simple because no data on regions and surfaces are necessary, respectively. A curved edge is apporximated by a polygonal line with straight edges practically. A curved surface is also approximated by small segment surfaces so that each of them seems practically flat. This modeling is applied to a digital terrain model.
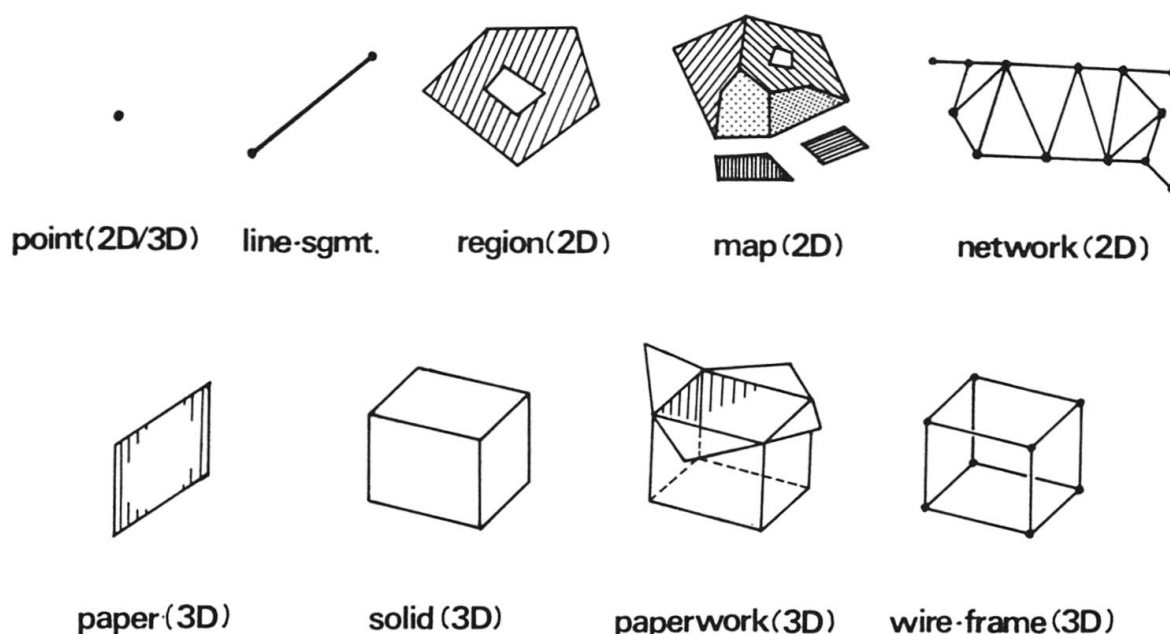
point(2D/3D)   line-sgmt.   region(2D)   map(2D)   network(2D)

paper(3D)   solid(3D)   paperwork(3D)   wire-frame(3D)

Fig.8 Geometrical Models

There are many ideas for programming techniques. Recently, a smaller program unit is becoming more covenient in order to use a micro-processor. Due to a physical limit of memory size, a large amount of data are managed as a data base out of the memories. Each program uses its own data structure to suit fast processing. Naturally, a pre-and-post processor is required to manage the data among a data base.

The principal aim of modeling is to save the efforts for data preparation. A problem of automatic mesh generation is from such a requirement for an FEM network. It is the best way to ask a computer by a language with a few input data. The author proposes a language GEOMETRY for the processing of 2D and 3D shapes. The language manages a shape as a figure defined in the previous article.

4.3 Numerical Properties of Figures

a) Area:
   The area of a region or a surface is a typical value frequently calculated. We know well how to calculate it about a fundamental figure such as a triangle, a rectangle or a circle. Any polygonal shape is practically calculated as an assembly of partitioned triangles or rectangles. The area has a value with a sign either positive or negative. For example a hole has a negative value so that an exact value is obtained for a figure with holes.

   In order to calculate the area, the data structure of a figure should have the topological information on the edges, that is, the direction of an edge is positive when the corresponding region is on its left side. As to a 3D-surface, the area is represented by a vector whose size is the value of area. The direction of vector is normal to the surface indicating from inside to outside through the surface.

b) Centre of gravity:
   This is mathematically a mean of coordinate values of the points belonging to a figure. The centre of gravity of a triangle is a mean of its three vertices, however, that of a triangular ring becomes the centre of an inscribed circle. The cnetre of gravity of a solid material is calculated by an assembly of partitioned tetrahedrons, whose centres are each a mean of four vertices.

c) Static moment:
   This is an associated value with the above two properites, that is a product of both.

d) Moment of inertia and radius of gyration:
   Normally, a moment of inertia of a figure is calculated based on its local coordinate system whose origin is set at the centre of gravity. A value of moment inertia is algebraically the sum of quadratic product of two coordinate values. For this reason, a moment of inertia is defined as a symmetric matrix in our discussion. It is calculated by the summation of dyads on position vectors (see Tables-3/4).

   A radius of gyration for a sectional figure of a column is a root of a diagonal element of moment of inertia after dividing by its area. It is statistically a root mean square of coordinate values, namely, a standard deviation of coordinate values.

e) Principal axes:
   When a beam is bent by a bending moment, it turns a little out of an expected bending plane if the sectional figure of a beam is asymmetric against axes. Principal axes tell the new local coordinate system by which the moment of inertia becomes a diagonal matrix.
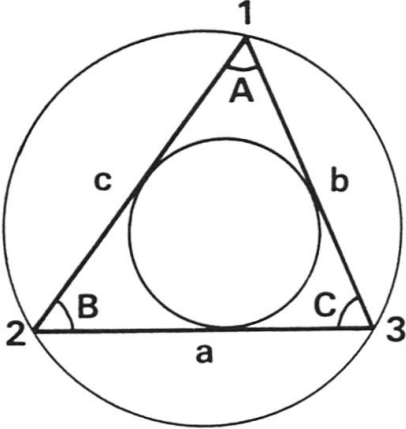
Table-3 Parameters for a Triangle
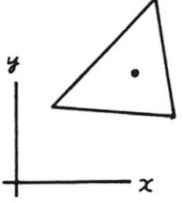
	Area; $F = s(s-a)(s-b)(s-c)$ where, $s = (a+b+c)/2$ Angles; $\sin(A) = 2F/bc$, $\cos(A) = (S-a^2)/bc$ $\sin(B) = 2F/ca$, $\cos(B) = (S-b^2)/ca$ $\sin(C) = 2F/ab$, $\cos(C) = (S-c^2)/ab$ where, $S = (a^2+b^2+c^2)/2$ Radius of inscribed circle R_i; $R_i = F/s$ Radius of circumscribed circle Ro; $Ro = abc / 4F$
Centres of a Triangles; $r = k_1 \cdot r_1 + k_2 \cdot r_2 + k_3 \cdot r_3$ where, positions of vertices are vectors r_1, r_2 and r_3. parameters have a relation; $k_1 + k_2 + k_3 = 1$	
r_g	Centre of gravity; $k_1 = 1/3$ $k_2 = 1/3$ $k_3 = 1/3$
r_i	Centre of inscribed circle; $k_1 = a/2s$ $k_2 = b/2s$ $k_3 = c/2s$
r_o	Centre of circumscribed circle; $k_1 = a^2(S-a^2)/8F^2$ $k_2 = b^2(S-b^2)/8F^2$ $k_3 = c^2(S-c^2)/8F^2$
r_n	Centre of three normals intersected; $k_1 = (S-b^2)(S-c^2)/4F^2$ $k_2 = (S-c^2)(S-a^2)/4F^2$ $k_3 = (S-a^2)(S-b^2)/4F^2$
	Moment of inertia against WC; $J = \dfrac{F}{12}(r_1 r_1 + r_2 r_2 + r_3 r_3) + \dfrac{3F}{4} r_g r_g$ where J is a matrix, $r_i r_j$ is a dyad.

Table-4 Parameters for a Tetrahedron

Centre of gravity;

$$r_g = (r_1 + r_2 + r_3 + r_4)/4$$

Volume;

$$V = \frac{1}{6} \begin{vmatrix} r_2 - r_1 \\ r_3 - r_1 \\ r_4 - r_1 \end{vmatrix}$$

(a determinant of a 3x3 matrix)

Moment of inertia;

$$J = \frac{V}{20} (r_1 r_1 + r_2 r_2 + r_3 r_3 + r_4 r_4 + 16 r_g r_g)$$

where J is a 3x3 matrix and

$r_i r_i$ is a dyad.

A mathematical procedure to find the principal axes is to compute eigen vectors as well as eigen values of a given symmetric matrix. From a geometrical point of view, however, the choice of direction of eigen vectors is important. Eigen vectors should be arranged so as to become new coordinate axes accounting of each corresponding old axis.

f) Volume of a body:
A solid figure is assumed as an assembly of cones with each polygonal base. After computing the area of base, an elementary formula for a cone is applied to obtain the volume. As described in the item of area, the sign of volume depends on the direction of the area vector. An edge of a solid surface should have the information on both surfaces. For this reason, the data structure of a solid figure becomes more complicated than that of a network frame structure.

g) Radii and centres:
A circumscribed circle or a inscribed circle of a triangle is well known in the elementary geometry. It becomes a little difficult by an analytical equation. Table-3 summarizes those parameters for a triangle.
As to a tetrahedron, a circumscribed or a inscribed sphere has a geometrical interest. It is, however, difficult to show a refined practical equation. An aim of the computational geometry is found on such a problem. To decide the centre of an inscribed sphere seems simple by the aid of descriptive geometry, but becomes difficult for numerical evaluation without the aid of a computer.

h) Package size:
A circumscribed rectangle which contains a given 2D-figure is useful rather than to compute a circumscribed circle. It helps for instance to decide a graphical window where an aiming figure stays reasonably.

A maximum or minimum stress due to a bending moment is calculated on a sectional figure of a beam accounting with the centre of gravity and the package size.

A circumscribed cube is also used to know the package size of a given 3D-figure. Such a package size is frequently used during geometrical calculation without any special cares.

4.4 Logical Operations between Figures

Addition of two triangles, in a common sense, means that two triangles are connected side by side making a polygonal shape. There are no algebraical rules similarly applied to such calculation. Instead of addition, we say two triangles are operated by a logical OR. A figure is assumed as a system of points, namely, a set of points. The logical operation is carried out between two sets of points by the Boolean algegra.

In order to understand the set theory among logical values, a Venn diagram is used as a tutorial tool (Table-5). Since a figure is a set of points in our discussion, Table-5 classifies how two figures make an interference causing another figure. Supposing the same material, logical OR operates as welding of two figures, while one of two common parts, given by logical AND, is lost. Cutting off a part of material needs an opposit figure that represents a die.

Logical NOT produces an imaginary material which fills the whole space infinitely except the region where a figure has ever stayed. Instead of terms AND/OR/NOT more understandable words are recommended for a common user, for example, INT for intersection, UNI for unify, and SUB for subtraction. Each operation is carried out by one or more logical steps.

For the case of different materials, the interference between two figures causes several effects. If two figures are painted with different colours respectively, there are two possible ways to paint the common area. One figure is overlaid on another so as to hide a part of background colour. Instead of this, a mixed colour may be used to identify the common area. A graphic language to drive a colour graphic screen supports the overlay feature to make an image. Since a white colour is synthesized by three primary colours, no image will appear on a white background without replacing it.

Elimination of hidden lines is an important technique to obtain a realistic image of a 3D-figure on a graphic device. The problem has a difficulty due to logical operation among 2D-figures. A figure should have a data structure like a regional map. A figure is consisted of element figures each of which have some attributive data. After a logical interference is tested among figures, the data are compared so as to obtain a desired result. A relative distance is one of attributes to compare the overlaying depth to decide whether visible or not.
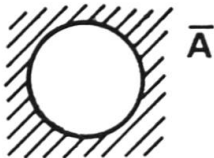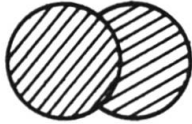
Three categories of interference among 2D-figures are shown in Table-5 as MAP, HID and CUT with respect to the processing of edges. A MAP image is obtained when all edges remain after logical operation. To unify regions by eliminating some edges is a procedure of HIDden line elimination. To set a new border CUTs a region into two parts.

An intersected point between two line segments is a result of the logical AND. Supposing an engineering drawing, the four resulted line segments should be identified whether or not they include the intersected point as a common point. A line segment at hidden side should be drawn so telling that the point is invisible. So that it will be drawn keeping a little distance from that point. This is the art of drawing to emphasize the line crossing.

A three dimensional map structure may be supposed like a honeycomb which has common surfaces separating local spaces. It is, however, out of our discussions.

Sometimes a status code is only required to tell the information whether two figures may cause intersect. Tables 1/2 show such status codes on geometrical elements. The status codes between two figures are defined almost the same but a little difference. As for a point against a region, for instance, exclusive(outside), inclusive(inside) and contact along an edge or a vertex are considered. One of difficult problems for the computational geometry is to find the similarity and congruency among figures. Such a classification suits for a man's eye with a reasonable man-machine communication.

Table-5 Logical operations between figures

Figures	Logical operations	Commands
	A and B are exclusive	
$\overline{A}$	NOT A (compliment of A)	EVERT A
A∩B	A AND B (logical AND)	INT A,B,... (product)
A∪B	A OR B (logical OR)	UNI A,B,... (addition)
	NOT(NOT(A) OR B)	SUB A,B,... (subtraction)
	A AND B, NOT(NOT(A) OR B) and NOT(NOT(B) OR A)	MAP A,B,... (mapping)
	A and NOT(NOT(B) OR A)	HID A,B,... (hidden lines)
	NOT(NOT(A) OR B) and A AND B	CUT A,B,... (cutting)

4.5 Computational Problems

To draw a triangle using three lengths is a fundamental problem in the descriptive geometry. To decide the coordinates of the vertices is a computaional problem which requires more numerical conditions. The algorithm is normally very simple and clear graphically. Sometimes, to compute a numerical value takes a lot of efforts notwithstanding a simple problem. The computational geometry aims at numerical processing in general. But in narrow sense, it deals with a problem related to so many points and lines that it becomes only practical by the aid of a computer.

Previous articles have discussed fundamental rules, that is, how to manage geometrical quantities from a theoretical viewpoint. Computational problems are to discuss the practical algorithm in a computer implementation. In order to talk about algorithm, however, a method of calculation should be mentioned so that the memory size and computing time are wihin a permissible limit for a computer. There may be therefore different methods to suit for either a big frame or a micro-processor. Several interesting topics are introduced in the following paragraphs. Each story may be too short to explain the practical techniques. Thus, a reader needs to grasp a preliminary knowledge before he meets each problem.

a) Generation of a curved line:
   A smoothly curved line is frequently required so as to connect given key positions. A part of a curved line between two key positions is a segment, for which a certain equation is employed taking into account its boundary conditions. A quadratic equation suits to represent a partially curved line because of its simplicity and wide adaptability. A third position is chosen in this case to decide the boundary conditions. Tables-6/7 summarize the quadratic curve generators. A parameter is shown there to select the line characteristics such as an elliptic, parabolic or hyperbolic line.

   A cubic equation makes a segment fit with two reference positions which are analyzed so as to satisfy the boundary conditions at both ends. This is well known as a third-order spline curve. A structural engineer will find this algorithm to be the same as that to compute the deformation of an elastic continuous beam supported at different height (Table-8).

b) Generation of a curved surface:
   A curved surface is usually approximated by curved lines in two directions. A partial curved surface is considered by a rectangular or a triangular mesh the edges of which are curved line segments. This is called a patch. Table-9 shows some curved surface equations for a proposed patch. In addition to key positions at the corner of a patch, some reference positions have to be decided along the curved edges and inside the patch if necessary. An equation for a patch is quadratic and/or cubic along two directions.

   In order to decide the reference positions, boundary conditions are taken into consideration. An elastic plate or a grillage structure may be assumed as a numerical model covering all the surfaces. Two absolute curved line systems are practically recommended by the aid of previous item. Any curved surface is reconstructed as a seqence of regular patch or mesh with each straight edges. This is due to the requirement for numerical processing. A mesh size is chosen so small as enough to approximate a flat plane. A triangular mesh suits to satisfy the local mesh with flatness.

c) Equi-distant curved lines:
   Let us suppose a beach line as a given curved line. The problem is to decide another curved lines being equi-distant one another on the sea or inland. The problem appears, for instance, to set both side lanes of a road, the centre line of which is given by a curved line. There is no mathematical formula in general except a use of a computer for numerical trial and error method.   (Fig.9).

d) Triangular mesh generation:
   A number of points are randomly distributing in 2D domain. A problem is to
   connect these points so as to form triangular meshes. Several ideas are
   considered not to generate a triangle with a very acute or obtuse angle.
   Man-machine interface is desirable in this problem because there exist many
   different solutions.

e) Voronoi diagrams:
   The same random points are considered as that of the above paragraph. Here
   is a problem to decide local domains subjected to each points. Let us sup-
   pose a regional map. The points inside the map represent railway stations.
   Inhabitants will go to the closest station to catch a train. So each station
   has a domain where inhabitants belong to. Border lines between domains are
   bisectors among the points. A resulted map is called a Voronoi diagram.
   (Fig.10).

f) Automatic Mesh Generation for FEM Models:
   Since an FEM model is a network graphically, a lot of preliminary work is
   required for geometrical and topological characteristics. An engineering
   structure has more or less a regularity in shape. A few data are enough to
   generate the rest of data. Numbering on nodes and edges is carried out by a
   little trained manner. But such a work is not always necessary when an
   appropriate graphic tool shows an overall characteristic at a glance.
   A network model covering continuous 2D or 3D materials is created by no
   regular ways. The automatic mesh generation is therefore demanded by a user
   to save his cumbersome work and to avoid mishandling of data. The above
   item d) and Table-9 are applied as one of algorithms.

   Accounting the use of a colour graphic screen, a recent idea is to create a
   model depending on the mesh of screen. A right rectangular mesh is best one.
   The maximum number of nodes has a limit depending on a screen accuracy. At
   the same time, an analytical theory is studied to meet with this model.

g) Packing and editing:
   Building blocks for children are packed in a box with the least size. To cut
   out several shapes from a large sheet needs a similar idea of packing.
   Editing a document is a sort of packing problems of character font together
   with photographs and tables. A mathematical block puzzle is investigated by
   an analyst using a computer. From an experience by a lot of practical
   packing problems, however, it has become clear that the best solution is
   achieved by man-machine communications.
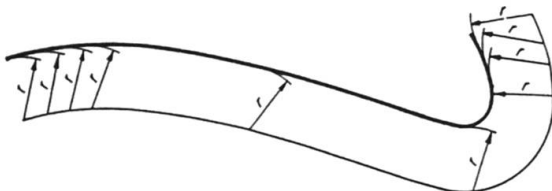
Fig.9 Equi-distant curved lines   Fig.10 A Voronoi diagram
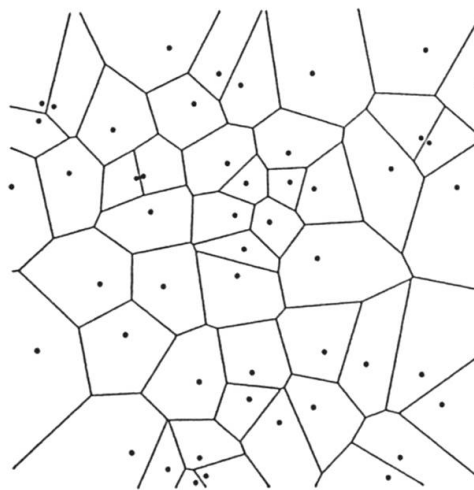
Table-6   Quadratic Curves through Three Points

General Equation;

$$r = \frac{1}{k_0}(k_1 \cdot r_1 + k_2 \cdot r_2 + k_3 \cdot r_3)$$

where,

$$k_1 = (1-t)(1-2t)$$
$$k_2 = t(4+a)(1-t)$$
$$k_3 = t(2t-1)$$
$$k_0 = k_1 + k_2 + k_3$$
$$\quad = 1 + at(1-t)$$

t: curve linear coordinate $(0,1)$
not proportional along length.

$a > 0$; hyperbola,
$a = 0$; parabola,
$a < 0$; ellipse.

hyperbola(a>0)

parabolla(a=0)

a quarter of an ellipse(a=-2+$\sqrt{2}$)

a half of an ellipse(a=-2)

two thirds of an ellipse(a=-3)
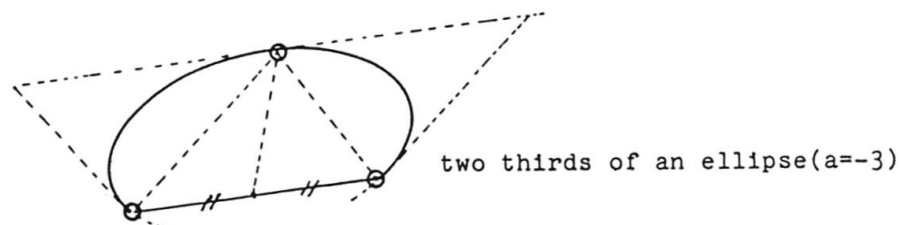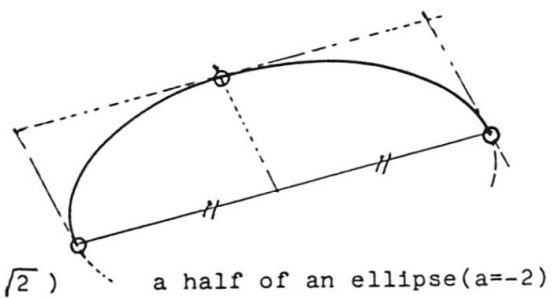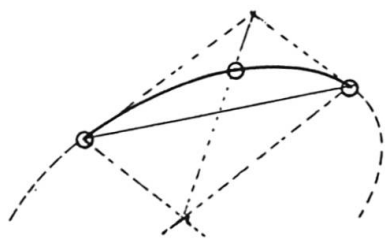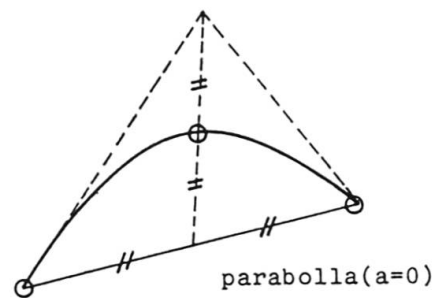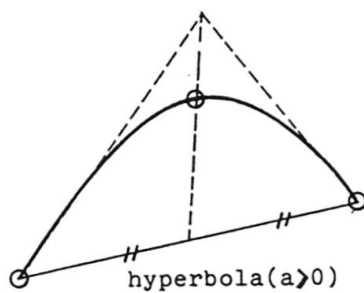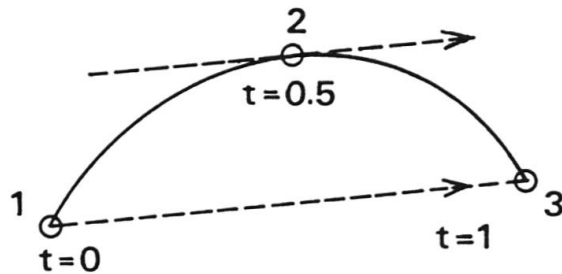
Table-7 Quadratic Curves being tangential to a Triangle

General Equation;

$$r = \frac{1}{k_0} (k_1 \cdot r_1 + k_2 \cdot r_2 + k_3 \cdot r_3)$$

where,

$$k_1 = (1-t)^2$$
$$k_2 = t(2+a)(1-t)$$
$$k_3 = t^2$$
$$k_0 = k_1 + k_2 + k_3$$
$$\quad = 1 + at(1-t)$$

t : curve linear coordinate $(0,1)$
not proportional along length.

$a > 0$; hyperbola,
$a = 0$; parabola,
$-2 < a < 0$; ellipse(inside),
$-4 < a < -2$; ellipse(outside).

$$h/H = (2+a)/(4+a)$$

hyperbola(a>0)

parabolla(a=0)

a quarter of an ellipse(a=-2+$\sqrt{2}$)

(a=-1)

two thirds of an ellipse(a=-3)

$$a = -4(\sin\beta)^2$$

$$\alpha = 2\beta$$

$$a' = -4+4(\sin\beta)^2$$

round arcs

three quarters of an ellipse(a=-2-$\sqrt{2}$)
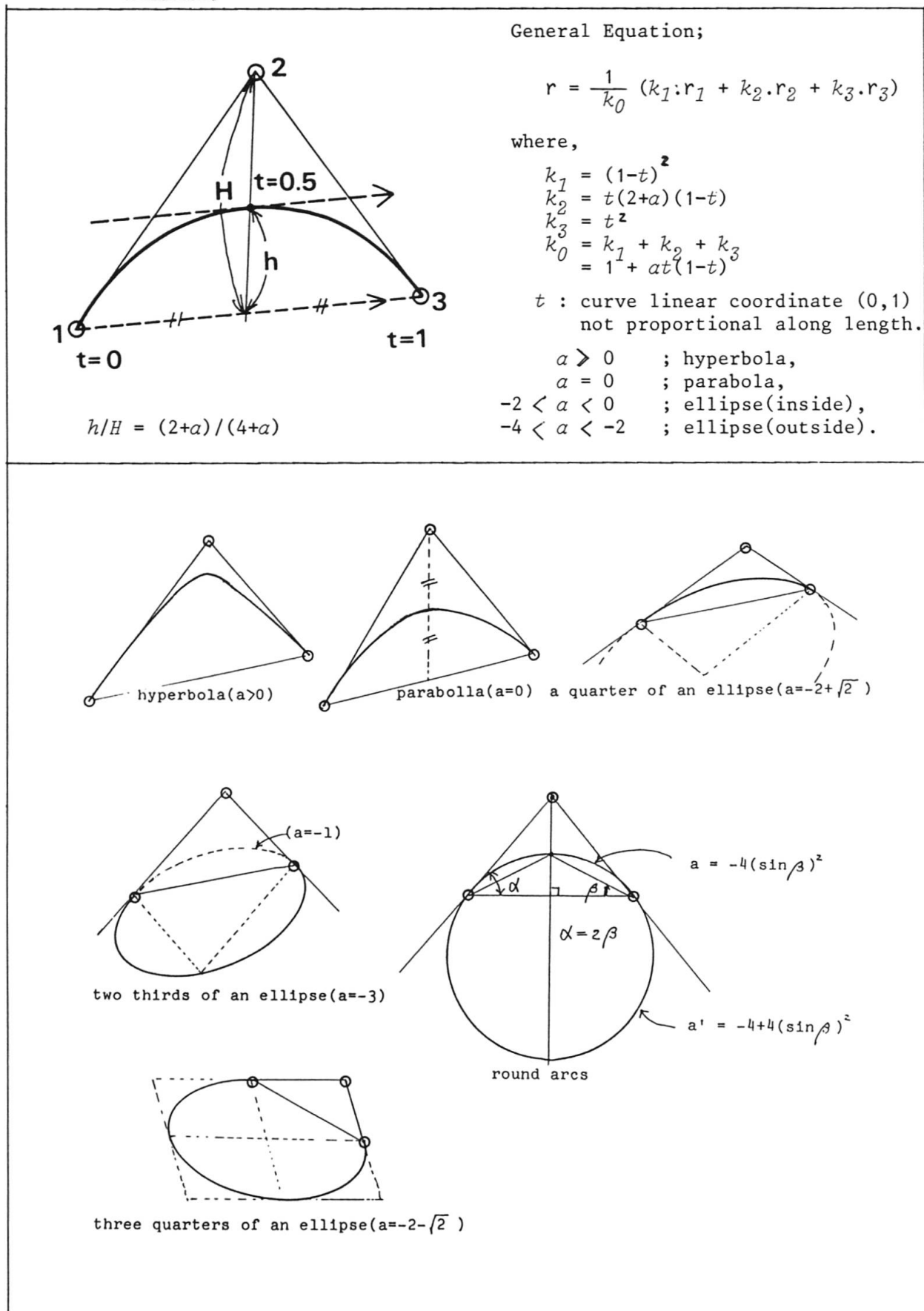
Table-8 Cubic Curves by given four Points

General Equation;

$$r = k_1.r_1 + k_2.r_2 + k_3.r_3 + k_4.r_4$$

where,

r, r_1, r_2, r_3 and r_4 are position vectors either 2D or 3D.

k_1, k_2, k_3 and k_4 are functions of t shown below.

t : curve linear coordinate between $(0,1)$.
It is not proportional along the length of line.

Boundary conditions are;

$t=0$: $r = r_1$
$t=1$: $r = r_4$

Two intermediate reference points r_2 and r_3 are used either on the
curve at $t=1/3$ and $t=2/3$, respectively, or to specify the tangential
lines at $t=0$ and $t=1$, respectively.

$t=1/3$; $r = r_2$ or $t=0$; $dr/dt = 3(r_2 - r_1)$
$t=2/3$; $r = r_3$ or $t=1$; $dr/dt = 3(r_4 - r_3)$

a)
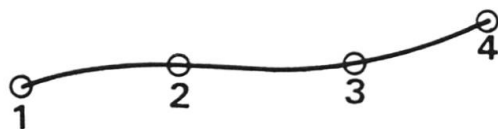
$k1 = (1/2)(1-t)(1-3t)(2-3t)$

$k2 = (9/2)t(1-t)(2-3t)$

$k3 = -(9/2)t(1-t)(1-3t)$

$k4 = (1/2)t(1-3t)(2-3t)$

b)
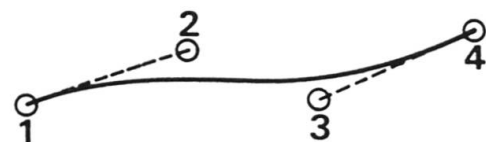
$k1 = (1-t)^3$

$k2 = 3t(1-t)^2$

$k3 = 3t^2(1-t)$

$k4 = t^3$

c)
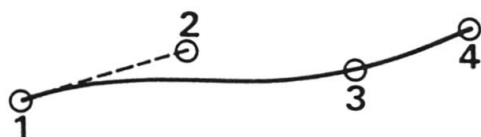
$k1 = (1/4)(1-t)(2-t)(2-3t)$

$k2 = (3/2)t(1-t)(2-3t)$

$k3 = (27/4)t^2(1-t)$

$k4 = t^2(3t-2)$

d)
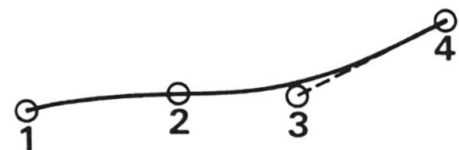
$k1 = (1-t)^2(1-3t)$

$k2 = (27/4)t(1-t)^2$

$k3 = -(3/2)t(1-t)(1-3t)$

$k4 = -(1/4)t(1+t)(1-3t)$

Table-9 Curved Surfaces covering Given Space Points

General Equation;

$$r = k_{11} \cdot r_{11} + k_{12} \cdot r_{12} + \ldots$$
$$+ k_{21} \cdot r_{21} + k_{22} \cdot r_{22} + \ldots$$
$$+ \ldots \quad \ldots \quad \ldots$$

where, k_{ij} : a function of two parametric coordinates u and v at a corresponding point number i and j.

i : either $i=1$, 2 and 3 at each $u=0$, 1/2 and 1, or $i=1$, 2, 3 and 4 at each $u=0$, 1/3, 2/3 and 1.

j : either $j=1$, 2 and 3 at each $v=0$, 1/2 and 1, or $j=1$, 2, 3 and 4 at each $v=0$, 1/3, 2/3 and 1.

u, v : parametric curve coordinates along two directions in the range of each (0,1).

---

a) Quadratic Curved Surface covering 9 Points;

$$k_{ij} = f_i(u) \cdot f_j(v)$$

$$f1(t) = (1-t)(1-2t)$$
$$f2(t) = 4t(1-t)$$
$$f3(t) = -t(1-2t)$$

---

b) Quadratic Curved Surface by Given 8 Points along Edges;

$$k11 = (1-u)(1-v)(1-2u-2v)$$
$$k12 = 4(1-u)v(1-v)$$
$$k13 = (1-u)v(2v-2u-1)$$
$$k21 = 4u(1-u)(1-v)$$
$$k22 = 0$$
$$k23 = 4u(1-u)v$$
$$k31 = u(1-v)(2u-2v-1)$$
$$k32 = 4uv(1-v)$$
$$k33 = uv(2u+2v-3)$$

---

c) Cubic Curved Surface covering 16 Points;

$$k_{ij} = f_i(u) \cdot f_j(v)$$

$$f1(t) = (1/2)(1-t)(1-3t)(2-3t)$$
$$f2(t) = (9/2)t(1-t)(2-3t)$$
$$f3(t) = -(9/2)t(1-t)(1/3t)$$
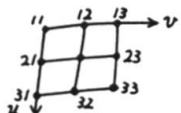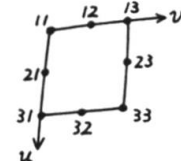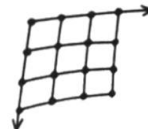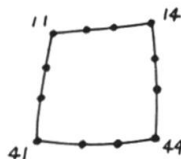$$f4(t) = (1/2)t(1-3t)(2-3t)$$

---

d) Cubic Curved Surface by Given 12 Points along Edges;

$$k11 = (1/2)(1-u)(1-v)(2-9u(1-u)-9v(1-v))$$
$$k12 = (9/2)(1-u)v(1-v)(2-3v)$$
$$k13 = -(9/2)(1-u)v(1-v)(1-3v)$$
$$k14 = (1/2)(1-u)v(2-9u(1-u)-9v(1-v))$$
$$k21 = (9/2)u(1-u)(2-3u)(1-v)$$
$$k22 = k23 = 0$$
$$k24 = (9/2)u(1-u)(2-3u)v$$
$$k31 = (9/2)u(1-u)(1-3u)(1-v)$$
$$k32 = k33 = 0$$
$$k34 = -(9/2)u(1-u)(1-3u)v$$
$$k41 = (1/2)u(1-v)(2-9u(1-u)-9v(1-v))$$
$$k42 = (9/2)uv(1-v)(2-3v)$$
$$k43 = -(9/2)uv(1-v)(1-3v)$$
$$k44 = (1/2)uv(2-9u(1-u)-9v(1-v))$$

---

e) Quadratic Triangular Curved Surface by Given 6 Points;
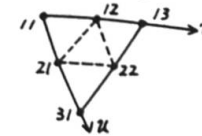
$$k11 = w(2w-1), \quad \text{where } w=(1-u-v)$$
$$k12 = 4vw$$
$$k13 = v(2v-1)$$
$$k21 = 4uw$$
$$k22 = 4uv$$
$$k31 = u(2u-1)$$

---

f) Cubic Triangular Curved Surface by Given 10 Points;
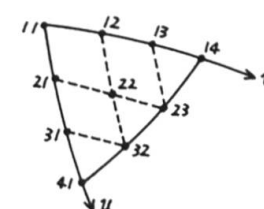
$$k11 = (1/2)w(3w-1)(3w-2)$$
$$k12 = (9/2)vw(3w-1)$$
$$k13 = (9/2)vw(3v-1)$$
$$k14 = (1/2)v(3v-1)(3v-2)$$
$$k21 = (9/2)uw(3w-1)$$
$$k22 = -27uvw$$
$$k23 = (9/2)uv(3v-1)$$
$$k31 = (9/2)uw(3u-1)$$
$$k32 = (9/2)uv(3u-1)$$
$$k41 = (1/2)u(3u-1)(3u-2)$$

---

g) Cubic Triangular Curved Surface by Given 9 Points along Edges;
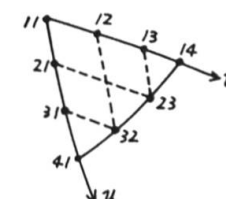
$$k11 = (1/2)w(3w-1)(3w-2)$$
$$k12 = (9/2)vw(2w-v)$$
$$k13 = (9/2)vw(2v-w)$$
$$k14 = (1/2)v(3v-1)(3v-2)$$
$$k21 = (9/2)uw(2w-u)$$
$$k22 = 0$$
$$k23 = (9/2)uv(2v-u)$$
$$k31 = (9/2)uw(2u-w)$$
$$k32 = (9/2)uv(2u-v)$$
$$k41 = (1/2)u(3u-1)(3u-2)$$

---

h) Linear Synthesizing from 4 Corner Points and 4 Edge Curved Lines;

$$r = (1-u) \cdot q_{13}(v) + u \cdot q_{24}(v) + (1-v) \cdot q_{12}(u) + v \cdot q_{34}(u)$$
$$-(1-u)(1-v) \cdot r_1 - u(1-v) \cdot r_2 - v(1-u) \cdot r_3 - uv \cdot r_4$$
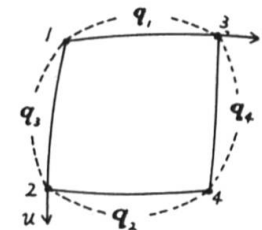
where,

r_i : corner point in vector.

$q_{ij}(t)$: a curved line from point i to point j as shown in Tables 6/8. Any relation is applied whenever it satisfies;

$$t=0 : q_{ij}=r_i$$
$$t=1 : q_{ij}=r_j$$

5. IMAGE TRANSFORMATION

5.1 Images

A man who imagines a geometrical shape in his mind must decompose it into a set of points and line segments. These data are sent to a graphic device while some attributes on colour, line species, or line width are transmitted. An image is physically consisted of points and line segments. There are two kinds of graphic systems for making a line segment. A pen plotter is able to draw a line. On the other hand, a line is practically constructed by a series of points on a colour graphic CRT. Smoothness of a curve and fineness of texture have a practical limit due to the accuracy of a graphic device.

Illustrated images are more or less illusions which make some meaningful association in our minds. They happen to fall into a puzzle or something like a graphic art unless carefully designed. Notwithstanding his intents, another person may understand the shape as a quite different object. Specifications for engineering drawings therefore establish the rules to transform figures into images. On the other hand, characters are understandable regardless of any styling.

An image is created by the data of 2D-figure or 3D-figure after camera works. Vertices and edges are respectively transformed into points and line segments on a screen, however, regions or surfaces need special art of drawing to identify them among others. Hatching line, filling area by pattern, painting by colour are such techniquies. An image needs many non-geometrical properties such as colour, line width, line species and so forth.

5.2 Camera

The best way to understand an image transformation is to suppose a photograph processing. Let us suppose a camera looking at a 2D-figure in the right direction. Since a camera has a limited scope, we set a camera so that a figure comes inside a viewing finder reasonably. A rectangle area of a finder is called a 'window' which bounds the visible area on a plane where a figure stays. As for a 3D-figure, a conical space is visible through the window.

A shot is taken onto a film with an appropriate size. A graphic device is an apparatus to visualize an image on a given device frame. The film is projected on the frame. The size and aspect ratio of the frame are proper characteristics depending on a device. The size of frame is therefore considered a unit square for numerical processing so as to meet with any graphic device. A partial rectangular area on the frame is called a 'viewport' where a film is projected and the rest is not exposed.

A film represents a data set for images. A user is, however, responsible to prepare the data by the aid of a graphical language. The data are directly transmitted to a graphic device, or stored temporarily in a file. A knowledge on an imaginary camera helps to prepare a fine film. A camera represents a user's eye. It stands at a certain position and looks toward some direction. Setting a camera is to decide a local coordinate system through which a view is measured.

A familiar 35mm camera is an apparatus to make a picture by the principle of point projection. A visible world depends on the species of lens such as wide, standard or telescopic one. Instead of focus length, an angle of scope becomes a common parameter regardless of any film size(Table-10). The parallel projection stands at the theoretical limit of a very long shot by a telescopic lens. Since an angle of scope becomes small, a window should be decided by the visible size in WC. After the definition of a camera, a user needs to prepares the data of figures in their original world coordinate system. When he drives directly a graphic device, a proportional world coordinate system is assumed on a graphic frame.

5.3 Coordinate Systems

A graphic device has generally a rectangular area in which a device coordinate system (DC) is defined for a pen positioning. The DC is measured by integer values whose range covers the physical size. A normalized device coordinate system (NDC) is therefore recommended regardless of any device species. The NDC has a square range (0,1)x(0,1), however it has an unused range either on top or right when the device frame is a rectangle.

A 2D world coordinate system(WC) is assumed to be equally overlaid with NDC at the first time. A 2D-figure may stay out of the device range. A declaration of a window activates the coordinate transformation so that the window is projected onto the full frame of the device. A declaration of viewport defines a local viewport for layouting several images on the same frame without interference among them.

A user prepares the data of 2D-figure based on the world coordinate system. He should remember his declaration on window and/or viewport, but he needs not be careful for the sizes dependent on a device. Enlarging, moving or rotating of images is possible in two different ways. One is carried out against a figure in WC. Another way is carried out against the data set of an image. Normally, the graphical language supports the latter way as if an image was a film before a projector.

The projection of a 3D-figure is designed in the same manner as that of the 2D-figure, but a window is three dimensional. A graphic language is not yet so kind to transmit a 2D-image on a screen that a user has to solve the following two problems. One is a projective transformation and the other is a hidden line and/or surface elimination.

5.4 Perspective Transformation

In order to visualize a 3D-figure, a camera represents a local coordinate system through which a position is measured. A point(x,y,z) in WC is transformed in the first step to a virtural point(x',y',z') in a local coordinate system of a camera. A view direction is assumed to be parallel to $-x'$ direction in our discussion. The values (y',z') are used for a coordinate of image on a parallel projection.

A transparent screen is supposed when a point projection is concerned. The screen is laid at $x'=-f$ before the origin of LC, perpendicularly to x' axis. Then a projected point on the screen is meausred as;

$$x'' = -f \quad ,$$
$$y'' = -f \, y'/ \, x' \quad \text{and}$$
$$z'' = -f \, z'/ \, x' \qquad (18)$$

where f represents the focus length of an imaginary camera. A set of (y'',z'') is a coordinate of a transformed 2D-figure. The relative size of the transparent screen againt the focus length is reduced into a parameter TH2 in Table-10.
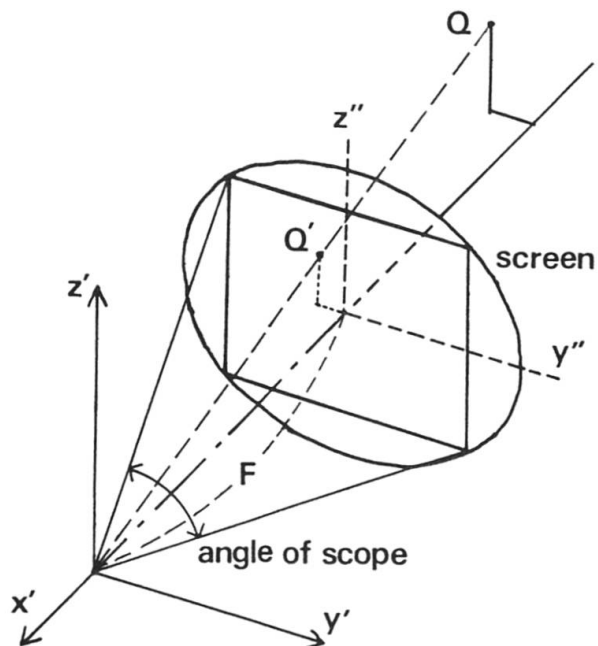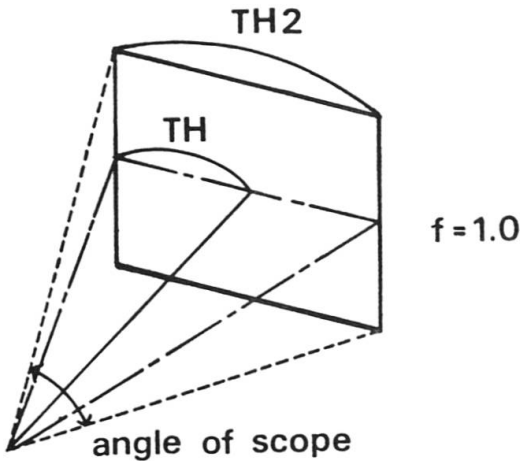
Fig.11 Perspective projection

Table-10 Camera Constants using a 35mm film

lens	focus f(mm)	angle of scope (degree)	2.tan(ang/2) TH2	
wide	18	100	1.983	
	24	83	1.472	
	28	75	1.277	
	35	63	1.020	
	40	57	0.904	
standard	50	47	0.724	
	85	29	0.430	
	100	24	0.354	
	135	18	0.264	
	200	12	0.175	
telescope	300	8	0.116	
	400	6	0.087	
	600	4	0.058	
	1000	2.5	0.036	

5.5 Hidden Lines and/or Surfaces Elimination

A realistic image of a 3D-figure is achieved by eliminating invisible lines and/or surfaces. This is essentially a three dimensional problem taking into a consideration about the distance between an eye and an object. Therefore, an original 3D-figure should have a correct data structure. As to a polyhedron, no invisible surfaces are necessary for the processing. A convex polyhedron is easy for making a view only showing visible surfaces. Whether visible or invisible depends on the plane equation of a surface. When an eye stays at its positive side, this surface is visible.

The hidden line phenomenon occurs at a countour edge which is bounded by a pair of a visible and invisible surfaces. A silhouette of a polyhedron is obtained by such edges that could make interference with another edges in the behind. A concave polyhedron has some edges connecting respectively a pair of surfaces in concavity. These edges are invisible whenever at least one of both surfaces is invisible. Such contour edges may stay over the opposit visible surfaces.

An edge should then be cut when it intersects with a contour edge in 2D domain. Its invisible part is in the silhouette side. Sometimes, a silhouette hides another silhouette. This is classified as hidden surfaces because no intersect will be found with contour edges but the potentially visible edges should be eliminated.

The algorithm of hidden line eliminatin is as mentioned in above paragraphs. It is practically useful on line drawings of polyhedrons, however, it has some incompleteness for partially hidden surfaces. In order to emphasize a stereo graphical effect, shading and contrasting may be drawn on the surfaces by means of line hatching, pattern filling or colour painting. A map structure is then required for an image. The contour edges, which may overlay on other surfaces, should be checked whether or not they become new borders in the map structure.

However, the procedure of hidden line elimination becomes very simple when a colour graphic CRT is available. Similarly to an oil painting on a canvas, a new painting replaces background colours. Beginning from a distant view, the painting procedure hides unnecessary surfaces automatically. This procedure works well prctically but could cause a little incompleteness by a lot of concavities among surfaces.

Notwithstanding a colour graphic device becoming convenient, engineering documents still use a lot of line drawings in monochrome. Dimensioning lines are, for instance, partly removed not to make interference among them or with a drawn shape. An art of drawing is important in a sense of engineering drawings. This is a sort of hidden line elimination that needs a lot of man-machine communication.

5.6 Graphical Language

Since a graphical device is physically controled by machine dependent codes, a user is kindly demanded to use a graphical language which is supplied by a machine fabricator. There have been many such languages with respect to object devices. It is required therefore to establish a common language regardless of any device species. The Graphical Kernel System(GKS) is now under preparation internationally. The GKS covers the whole specifications for the design of hardwares as well as for the control methods, namely a language. Because of the international standard, the GKS aims at rather a professional target than user friendliness.

A graphical language is basically a tool for the image processing, and not for the figure processing by the author's opinion. A lot of application softwares are required on the art of drawings. A rectangle drawing, for instance, has several possible methods. Conditions are; four corner positions, three corner positions, two base line positions plus height, two diagonal positions with parallel edges to the coordinate axes, left bottom point plus width and height, a middle point plus width and height, and so forth. An application software will impose one of such conditions.

The GKS never aims the standardized art of drawings. It is alike to prepare the painting materials for an artist. An automated engineering drawings will be classified as one of the arts. Some people have, however, expected that the GKS will set up a standard language for engineering documents. An engineer has to set up a lot of application softwares by the aid of the GKS. Every engineering field has its own special graphical target. Pre-and-post processor of structural design, such processor for FEM, landsacape modeling, use of aerial photograph, regional design by the aid of computer maps, automated documentation including computer made graphs, and so forth have interesting topics for application softwares.

6 LANGUAGE GEOMETRY

6.1 Introduction

Problems of geometry are usually analyzed by the mental association while looking at some analogical sketches. A few words are enough to tell a property with an appropriate sketch. For an instance, let us imagine to teach a way for a stranger. A use of a map is the best. Sometimes we have to teach him by the lingual aid. However, the correct information is always necessary on distance and direction, because he is a computer in our discussion.

The language GEOMETRY is designed to describe the geometrical conditions by which both a man and a computer can understand a problem without any sketches. A computer is able to aid a man graphically and furthermore to help him for numerical results. GEOMETRY is essentially a set of FORTRAN subprograms for the sake of wide adaptability to any computer systems. The subprograms are useful for a professional programmer. However, a common user, possibly a student, is invited to use an interactive language NUCE-BASIC under which GEOMETRY is managed.

6.2 Workability

Since a BASIC language is becoming a popular tool, the author composed a BASIC interpretor by FORTRAN, say NUCE-BASIC. It works samely as a BASIC wihch is adopted in a micro computer. NUCE-BASIC is designed to be able to define more additional commands which represent FORTRAN sub-programs. GEOMETRY is used under such a circumstance as special commands. Several subsets of GEOMETRY are prepared in order to suit for special field of engineering. Graphic softwares are supported in every case.

The final aim of CAD (the computer aided design) is generally to obtain the engineering documents for a customer. On the contrary, CAM (the computer aided manufacturing) extends a target up to a machinery processing. During the design works at a computer terminal, an engineer always tries several possibilities to find a reasonable solution. A quick response is best in these trials and no documents of good quality are required yet until he will be satisfied on a result. A historical record, namely a log-file, is helpful to trace back and to play back his works. The language GEOMETRY is co constructed under NUCE-BASIC.

Following articles show several topics relating to the language GEOMETRY.

6.3 A Tool for Analytical Geometry

Let two lines be x+y−5=0 and 3x−2y+1=0; and compute an intersected point P of this two lines. Further, let another point Q be (0.2, 0.2); and define a line equation that connects two points P and Q.......

The above proposition is a very simple problem. Figure-12 shows its graphic solution as well as BASIC composition. Five arrays with size 3 are declared to represent lines, two of which are coordinate axes. A line is considered here by Eq.4. A graphic command DPLINE is designed to display a line on a given drawing frame.


```
> LIST
    10 REM --- GRAPHIC DEVICE
    20    DPOPEN : DCWIND 1., 2., 20.
    30 REM --- SPECIFY ARRAYS
    40    DIM XL(3), YL(3)
    50    DIM AL(3), BL(3), PQ(3), P(2), Q(2)
    60 REM --- DRAW X-Y AXIS
    70    READ XL : DATA 1. 0. 0.
    80    READ YL : DATA 0. 1. 0.
    90    DPLINE XL : DPLINE YL
   100 REM --- DRAW LINE-AL, LINE-BL
   110    READ AL : DATA 1. 1. -5.
   120    READ BL : DATA 3. -2. 1.
   130    DPLINE AL : DPLINE BL
   140 REM --- COMPUTE INTERSECTED POSITION
   150    V2PLL AL, BL, P, 1.E-4, IMODE
   160 REM --- GIVE A POSITION-Q
   170    READ Q : DATA 0.2 0.2
   180 REM --- DEFINE A LINE-PQ AND DRAW
   190    V2LPP P, Q, PQ
   200    DPLINE PQ
   210 REM --- END OF DISPLAY
   220    DPAUSE : DPCLOS
>
```
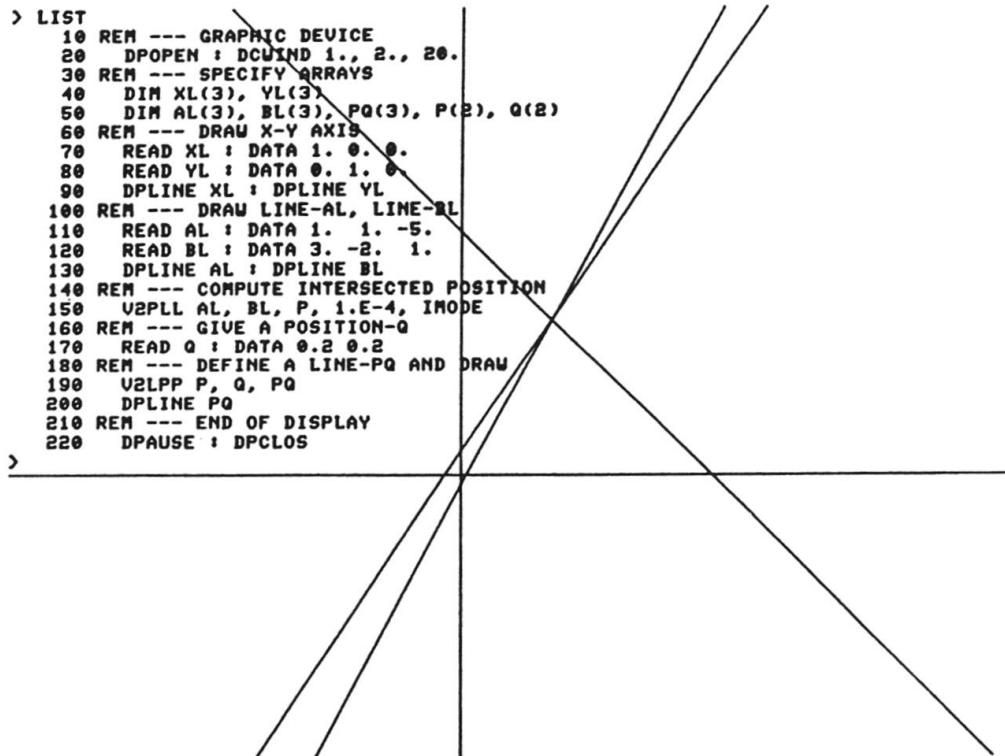
Fig.12 A graphical solution for an analytical geometry

6.4 Modeling of 3D-Figures

Solid-modeling is attracting a great interest for mechanical engineers with respect to CAD/CAM of mechanical parts. Structural engineers will find its application for the modeling of concrete structures. A very geomtrical example is shown in Fig-13.

A hexagonal cylinder is generated in the first step by a command PRCYLN. Two more similar shapes are created copying the original shape. Each cylinder is moved at a desired position. The three cylinders are logically unified to a single structure. Further, more three sets are copied again from the unified structure resulting twelve cylinders. After rotating them respectively, twelve cylinders are all connected. Then a camera is defined to look at this model. An image is obtained with hidden lines removed. An engineering interest is to obtain its volume, center of gravity and so forth. Nobody will try to compute such values without the aid of GEOMETRY.

```
LIST
    10 REM *** PUZZLE ***
    20 REM
    30 DPINT 1,2500,1800
    40 PRCYLN 'A', 6,, 2.0, 20.0
    50 PCOPY 'A', 'B', 'C'
    60 DIR(1)= 3.0 ! DIR(2)= 1.73 ! GDIS 'A', DIR
    70 DIR(1)=-3.0                 ! GDIS 'B', DIR
    80 DIR(1)= 0.0 ! DIR(2)=-3.464 ! GDIS 'C', DIR
    90 MUNI 'A', 'B', 'C'
    110 AXIS(2)=1.0 ! ANG=54.73*3.14159/180. ! CEN(1)=0.0
    120 GROTA 'A',AXIS,ANG,CEN
    130 PCOPY 'A','A2','A3','A4'
    140 AXIS(2)=0.0 ! AXIS(3)=1.0
    150 ANG=3.1416*0.5 ! GROTA 'A2',AXIS,ANG,CEN
    160 ANG=3.1416      ! GROTA 'A3',AXIS,ANG,CEN
    170 ANG=1.5*3.1416 ! GROTA 'A4',AXIS,ANG,CEN
    180 MUNI 'A','A2','A3','A4'
    190 MDCOMP 'A'
    200 CPOS(1)=30. ! CPOS(2)=30. ! CPOS(3)=20.
    210 TH=0.5
    220 DCAMO 'CAMERA',CPOS,TH
    230 DVIEW 'CAMERA'
    240 HIDDEN 'A','CAMERA'
    260 DPEND
>
```
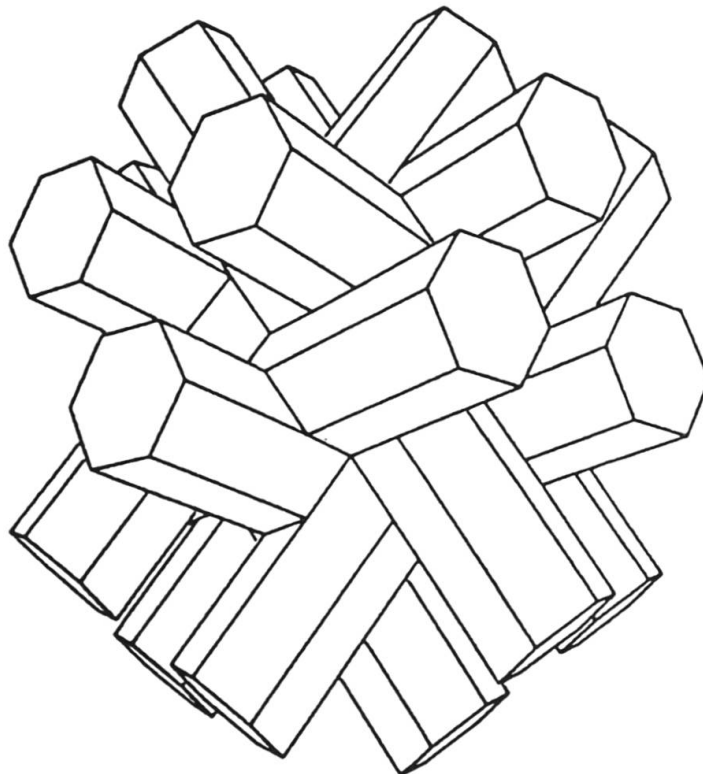

Fig.13 Generation and synthesizing solid models

7. CONCLUSIONS

Mathematical studies of geomety are discussed from a viewpoint in order to use a computer for teaching as well as practical using in structural engineering. Instead of hand writing on a sheet, a computer is able to show the geometrical information on a screen. The method to use a computer needs some special techniques, therefore a new kwyword computational geometry has appeard recently. A systematic consideration is then proposed on the geometrical properties before using a computer. Three classified categories are introduced, that is, geometrical element, figure and image.

Chapters 2 and 3 describe the analytical foundation on point, line and plane. The geometrical elements are analytical tools for which the vector and matrix calculation plays an important roll. A dyad, a sort of matrices, is introduced to synthesize a certain matrix for a numerical processing.

The modeling of a shape is discussed in chapter 3. A model is defined by an invisible 'figure' because it is a data set to describe a shape. Typical topics are mentioned there, however, practical procedures of modeling are summarized by commands in appendix-A. Some generators of figure are introduced such as for curved lines and curved surfaces.

Chapter 5 explains a technique to make a visible image from a data set of a figure. There are a lot of possibilities to make an image on a screen. An idea of a camera is helpful to understand the image transformation when we use a graphical language. A brief explanation is described on the problem of hidden line and/or surface elimination.

In order use a computer for geometrical problems, a lot of computer programs are required with reasonable user-friendlyness. A language GEOMETRY is proposed by the author in chapter 6. Names of subroutines are summarized in appendix-A as commands. These are designed to meet with the author's discussions of this paper. To make good workability on CAD, an interactive program NUCE-BASIC is recommended by the author. Some example procedures are shown in Figs.12 and 13.

Selected references are summarized in appendix-B.

Appendix-A   Selected Set of Commands in the language GEOMETRY

1. Definition of geometrical elements(2D/3D);

A line through two points	V2LPP/V3LPP
A line through a point with a direction	V2LPV/V3LPV
A line by intersection of two planes	V3L2F
Plane synthesizers	V3F3P/V3FPL/V3FPV

2. Interrelation between elements(2D/3D);

Distance between two points	V2DPP/V3DPP
Distance between a point and a line	V2DPL/V3DPL
Distance between a point and a plane	V3DPF
Length of a vector/normalization of a vector . .	VN21/VNR21/VN31/VNR31
A scalor product between two vectors(2D/3D/nD) . . .	VN22/VN32/VVPROD
Vecor product between two vectors	VN23/VVO3
An intersected point of two lines(2D)	V2PLL
An intersected point of a plane and a line(3D)	V3PFL
An intersected line of two planes	V3LFF

3. Vectors and Matrices;

Basic algebra for vectors	VMOVE/VPUSH/VADD/VSUB/VMULT/VSCAD
Basic algebra for matrices	VVDYAD/VMXPRD/VMPROD/VMXTRS/VMXMOV
Eigen values and eigen vectors for a symmetric matrix	VMEIG
Solution of linear equations	VMLEQ
Several statistical calculations .	VMEAN/VRMSQR/VGCEN/VDRIFT/VPEAK/VCLIP/VFFTR

4. Triangle and Tetrahedron;

An area and radii of a triangle by given 3 edges . . .	VTF3E/VTRI3E/VTRO3E
Centres of triangle	VTGCEN/VTICEN/VTOCEN/VTHCEN
Circumsphear/insphear of a tetrahedron by given 4 vertices .	V3R4PO/V3R4PI
Moment of inertia of a triangle/tetrahedron	VTMINT/V3J4PT

5. Moving, rotating, deformation, coordinate system;

Transformation matrices	V2ROTX/V3ROTX/V3ROTA
Procedures for transformation V2PROT/V2LROT/V2MROT/V3PROT/V3LROT/V3FROT/V3MROT	

6. Modeling of 3D-figures and processing;

Basic figure generator	PCUBE/PRCYLN/PRCONE/PREVL/PHEDRA
Synthesizers by logical operations	PUNI/PSUB/PINT/PCUT
Moving and rotating	PGDIS/PGROT/PTRANS
Display service	HEDISP/HLDISP/HFDISP/HIDDEN
Volume, axes and moment of inertia	PBODY

7. Modeling of 2D-figures and processing;

Basic figure generator	RRECT/RPOLYG/RFAN/RMAP
Synthesizer by logical operations	RUNI/RSUB/RINT/RCUT/RHIDE
Moving and rotating	RGDIS/RGROT/RGTRNS
Display service	HDISP
Area, axes and moment of inertia	RAREA

8. Display controls;

Initialization and termination	GKINIT/GKTERM/DPOPEN/DPCLOS
Device controls	DPAUSE/DPERAS/DPCOPY/DPCURS
Screen controls . DPWIND/DPVIEW/DPENSZ/DPENTX/DPENCL/DPCOLR/DPATRN/DPCHAR	
Drawing functions	DPMOVE/DPDRAW/DPTEXT/DPMARK
Basic image generators . . . . DPLINE/DPRECT/DPCIRC/DPARC/DPFAN/DPOLYG	
Application softwares	DHATCH/DCONTR/DSPLIN

Appendix-B   Selected Refefences and Some Notes.

(A) COMPUTATIONAL GEOMETRY is a recent keyword in the field of the graph theory and a network analysis. Typical works are found in the following reports.

1. Preparata,F,P: Steps into computational geometry. Tech.Rep. R-760, UELUENG - 77-2207, Coordinated Science Laboratory, Univ.of Illinois, Urbana, 1977.

2. Shamos,M.I.: Computational geometry, Disserttation, Yale Univ., 1978.

3. Brown,K.Q.: Geometric transforms for fast geometric algorithms.   Ph.D. Dissertation, Dep.of Computer Science, Carnegie-Melon Univ., 1979.

4. Shamos,M.I. and Hoey,D.: Closeset point problems.   Proc.of the 16th annual IEEE symposium on foundations of computer science.   New York, pp.161-162, 1975.

(B) DYADIC PRODUCT is scarecely introduced in most of books on geometry. It also never appears in any reports on numerical methods of matrices.

5. Shimada,S. :Mathematics in civil engineering(in japanese). Tokyo, 1967.

(C) MODELING OF 3D-SHAPES is found many in connection with CAD/CAM of mechanical parts.

6. Hosaka,M., Kimura,M. and Kakishita,N.: A unified method for processing polyhedra. Information Processing 74, pp.768-772, North-Holland, 1974.

7 Shimada,S. et al.:Researches on the development of computer programs subjected to the graphically aided design of structures. No.385132, Grant-in-Aid for Developmental Scientific Research, 1980 and No.00585131, 1982. (in japanese).

(D) GRAPHIC LANGUAGE

8. ISO/ANSI: ISO/TC97/SC5, Information Processing - Graphical Kernes System (GKS) - Functional Description. DP, 1982.

(E) AUTOMATIC MESH GENERATION

9. Frederick,C.O., Wong,Y.C., and Edge,F.W.: Two dimensional automatic mesh generation for structural analyses. Int.J.num.Meth.Engng.,2,pp.133-144,1970.

10. Zienkiewicz,O.C. and Philips,D.V.: An automatic mesh generation scheme for plane and curved surfaces by 'Isoparametric' coordinates. Int. J. num. Meth. Engng., 3, pp.519-528, 1971.

11. Durocher,L.L. and Gasper,A.: A versatile two-dimensional mesh generator with automatic bandwidth reduction. Computers & Structures, 10, pp.561-575, 1979.

12. Stefanou,G.D. and Syrmakezis,K.:   Automatic triangular mesh generation in flat plates for finite elements. Computer & Structures, 11,pp.439-464,1980.

13. Pissanetzky,S.: KUBIK;An automatic three-dimensional finite element mesh generator. Int.J.num.Meth.Engng., 17, pp.255-269, 1981.

14. Ghassemi,F.: Automatic mesh generation scheme for a two- or three-dimensional triangular curved surface. Computer & Structures, 15-6, pp.613-626,1982.

15. Nguyen-Van-Phai: Automatic mesh generation with tetrahedron elements. Int.J. num.Meth.Engng., 18, pp.273-289, 1982.