

A Survey of some Data-Dependent Criteria for
Triangular Tessellations using Fixed Nodes.

A. Malcolm

Numerical Analysis Report 1/90

Abstract

Classically triangular grid generation for numerical problems has largely been performed using the Delaunay criterion due to its min-max angle property which leads to well conditioned stiffness matrices in the finite element method. With the advent, however, of adaptive finite elements the emphasis has shifted to the generation of triangular grids which well represent initial data when this is interpolated upon them. To do this data-dependent mesh generating criteria have had to be devised. In this report we survey some such data-dependent criteria of Dyn, Rippa & Levin, contrasting them with another criteria based on equidistribution. Throughout we assume that a fixed set of nodes have been given on which to base the triangulation for the region.

1. INTRODUCTION

Classically, triangulations such as Delaunay [2] generate grids which produce well conditioned stiffness matrices when used with finite element methods, due to their max-min angle property, (Sibson [9]).

However, for many applications, for example producing initial grids for Moving Finite Elements [7] which moves the grid according to the data, the overriding consideration is accuracy of the data representation on the grid. This is not usually produced by the non-data dependent Delaunay triangulation [2] and so other criteria must be used. Dyn, Rippa and Levin [3] surveyed a number of criteria for generating a triangular grid from a given set of nodes to well represent a given underlying data function. Other techniques have also been employed, e.g. Sweby [10], who uses an approximate equidistribution technique based on the work of Carey & Dinh [1].

In this report we survey and compare these criteria as a first step towards producing a truly data dependent grid generation technique. Here we connect given nodes to produce triangular grids, while future work will involve adjusting nodal positions while keeping a fixed connectivity and eventually the two will be combined to produce a truly data dependent grid generation technique.

In Section 2 we outline the notation which will be used in the report, in Section 3 the concept of a data dependent triangulation is discussed, while the criteria based on such a concept are outlined in Section 4. In Section 5 we outline the procedures used in producing a data dependent triangulation. In Section 6 an outline is given of the testing procedure, the test functions and the data sets, while in Section 7 numerical and graphical results are produced. Finally a summary of the results and conclusions which can be drawn from them are presented in Section 8.

2. NOTATION

Before we consider the various criteria which we are investigating, we establish first the notation which will be used in the rest of the report.

Referring, where applicable to figure 1 we denote (following Dyn, Rippa & Levin [3]):

N is the number of data points, in the region Ω , and N_B is the number of points on the convex hull, Ω_B , of the region to be triangulated.

V is the set of data points, $v_i = (x_i, y_i)$, $i = 1..N$.

F is the Data Set. $F_i = f(x_i, y_i)$ at each data point where f is the actual function being worked on.

$t = 2(N-1) - N_B$ is the number of triangles, T_i , in a triangulation.

$e = 3(N-1) - N_B$ is the number of edges in a triangulation.

The triangulation, T , is the union of all triangles, so

$$T = \bigcup_{i=1}^t T_i, \quad T_i \cap T_j = \{\phi\}, \quad i \neq j.$$

f_T is the interpolating polynomial to the data on the triangulation, with

$$f_T = \bigcup_{i=1}^t f_i.$$

$$f_i = a_i x + b_i y + c_i, \quad \left[(x,y) \in T_i \right] \quad i = 1..t$$

where f_i is the local interpolating polynomial on triangle T_i .

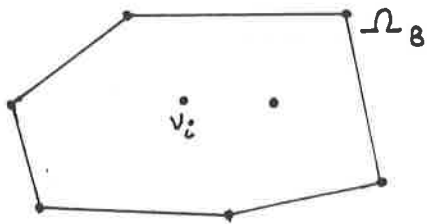
W is the array of the augmented set of data points.

$$W_i = (x_i, y_i, F_i), \quad i = 1..N.$$

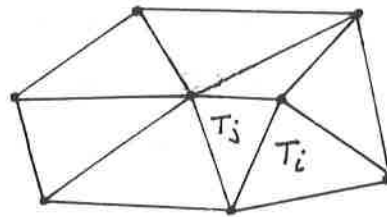
$\| \underline{z} \|$ is the Euclidean Norm of \underline{z} . i.e.

$$\| \underline{z} \| = \sqrt{z_1^2 + z_2^2 + \dots + z_m^2} \quad \text{where } \underline{z} \text{ is an } m\text{-vector.}$$

Since we shall be comparing different triangulations we need a notation to convey this. We denote that a triangulation T' is preferred to T in the sense of some cost function value (see below) by $T' < T$.



(i) convex hull of points



(ii) triangulation of points

Figures 1

The sense in which we judge triangulations is as follows, referring where necessary to figure 2

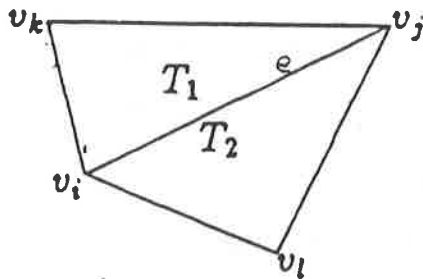


Figure 2. Two adjacent triangles.

Let $e = \overrightarrow{v_i v_j}$ be an internal edge of a triangulation T and let T_1 and T_2 be the two triangles sharing the common edge, e .

Suppose that $f_1 = P_1(x,y) = a_1x + b_1y + c_1$ $(x,y) \in T_1$

$f_2 = P_2(x,y) = a_2x + b_2y + c_2$ $(x,y) \in T_2$

are the linear interpolating polynomials on triangles T_1 and T_2 respectively.

For each interior edge e of the triangulation T , a real cost function $s = s(f_T, e)$ is assigned.

Let N and N' be real vectors of size q , with the elements ordered in a non-increasing manner. We can define ordering schemes on R^2 to say that $N < N'$ means that the triangulation T that produced N is better than the triangulation T' which produced N' , or $T < T'$.

The ordering schemes are:

- 1) ordering by the L_1 norm

$$R_1(N) = \sum_{i=1}^q |N_i|$$

and $N' \leq N$ if $R_1(N') \leq R_1(N)$

- 2) ordering by the L_2 norm

$$R_2(N) = \left[\sum_{i=1}^q |N_i|^2 \right]^{1/2}$$

and $N' \leq N$ if $R_2(N') \leq R_2(N)$

- 3) ordering lexicographically

$N \leq N'$ if the vector N is lexicographically not greater than N'

i.e. compare element by element in order 1, 2, ... q .

if $N_1 < N'_1$ then $N < N'$

if $N_1 > N'_1$ then $N > N'$

otherwise check $N_2, N'_2, N_3, N'_3, \dots, N_q, N'_q$,

if $N_i = N'_i, i = 1, \dots, q$ then $N = N'$.

Thus we now have methods of determining an ordering for a set of vectors.

In the next section we look at the concept of data dependent triangulations, and how we can define an optimal triangulation for a given criteria now that an ordering of triangulations is possible.

3. DATA DEPENDENT TRIANGULATIONS

In this section we define what is meant by an optimal data dependent triangulation and describe the procedure which was used in our attempts to produce such a triangulation.

We use triangulation criteria to order several triangulations thus enabling us to choose a preferred triangulation. We begin, following Dyn, Rippla and Levin [3], by defining what we mean by an optimal triangulation.

Definition 3.1

An optimal triangulation of a region Ω , given a fixed set of nodes, with respect to a given criterion is the triangulation T^* such that

$$T^* \leq T$$

for every triangulation T of Ω .

For this definition it can be seen that the optimal triangulation with respect to a criteria need not be unique.

An optimal triangulation of Ω always exists since there are a finite number of triangulations of Ω . However, it may be difficult to obtain this optimal triangulation in practice, and we may find ourselves in the situation of a local minimum of the cost function associated with the criterion.

Let T be a triangulation, e an internal edge of T and Q a quadrilateral formed by the two triangles having e as a common edge.

If Q is strictly convex then there are two possible ways of triangulating Q (see figure 3).



Figure 3. The two possible triangulations of a convex quadrilateral.

Definition 3.2.

An edge e is called locally optimal if $T \leq T'$ where T' is obtained from T by replacing e by the other diagonal of Q .

Definition 3.3.

A locally optimal triangulation of Ω is a triangulation T' in which all edges are locally optimal.

A data dependent triangulation, of a fixed set V , depends on the data vector F and so the preferred triangulation is not the same for all data vectors.

The optimization procedure used to construct the triangulations is based on the Local Optimization Procedure (LOP) suggested by Lawson [6], which can be written as follows:-

- (1) Construct an initial triangulation $T^{(0)}$ of Ω and set $T \leftarrow T^{(0)}$.
- (2) If T is locally optimal - end the procedure, else go to step 3.
- (3) Let e be an internal edge of T which is not locally optimal and let Q be the strict convex quadrilateral formed by the two triangles having common edge e .

- (a) Swap diagonals of Q , replace e by the other diagonal of Q , therefore transforming T to T' .
- (b) Set $T \leftarrow T'$ and go to step 2.

This means that after every edge swap occurs, the resulting triangulation is strictly lower in the ordering than the previous one. Since the number of triangulations is finite then the LOP converges after a finite number of edge swaps to a locally optimal triangulation.

In the next section we shall describe the various data dependent criteria, the results of which we shall later compare.

4. CRITERIA FOR PRODUCING TRIANGULATIONS

The criteria are taken in a systematic order. First we look at the most common non-data dependent criteria which are Delaunay and Minimum Weight, and then we look at data-dependent extensions of these criteria. We then look at the data dependent criteria categories, nearly C^1 (NC1) and near planar, from Dyn, Rippa & Levin [3]. Finally we look at a criteria based on approximate equidistribution, (Sweby [10]).

(a) Non-Dependent Criteria

(1) Delaunay Triangulation

Delaunay [2] is the most common triangulation technique used since it produces "good", regular, almost equiangular grids which possess favourable properties when used with the Finite Element Method. The Delaunay Criteria seeks to maximise the minimum angle in the triangulation, and it has been shown that an equivalent property is that the circumcircle of each triangular element contains no nodes in its interior, (Sibson [9]). This is demonstrated in Figure 4.

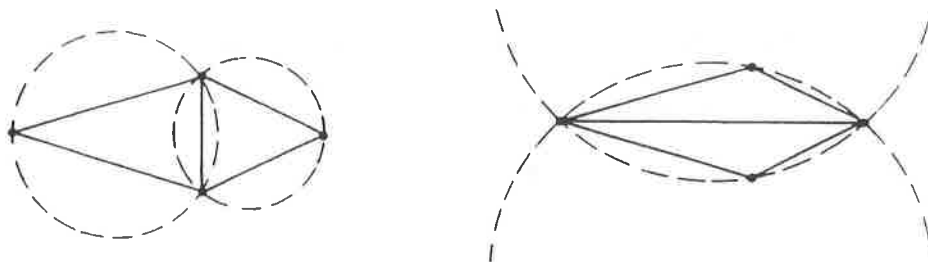


Figure 4. Example of Delaunay and non-Delaunay triangulation.

Delaunay triangulation can be achieved using either an insertion polygon or a diagonal swapping process. Each process is started by the introduction of 3 or more artificial nodes which are positioned well away from the convex hull Ω_B of the points to be triangulated (see Figure 5).

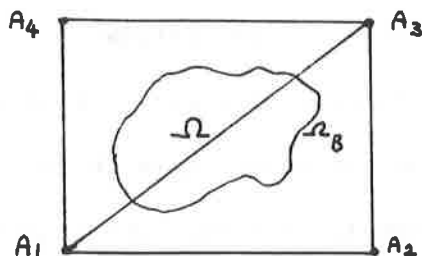


Figure 5. A_1, A_2, A_3, A_4 are artificial nodes.

The artificial nodes are triangulated and then the nodes of V are introduced one by one. As each node is inserted the grid is retriangulated. In the insertion polygon scheme, if the new node lies inside an element circumcircle (i.e. the circle passing through its vertices), then the element is removed. When all such elements have been removed we are left with a polygon, the insertion polygon. The new node is then connected to all points on the boundary of this insertion polygon to give the new triangulation (Mock [8], Hunt [5]).

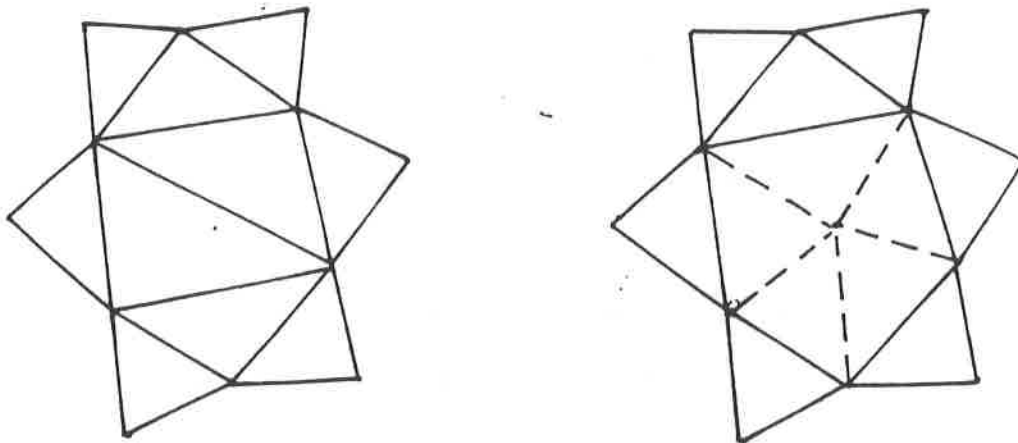


Figure 6. Node insertion by the insertion polygon method.

In the diagonal swapping procedure, the inserted node is connected to the vertices of the element it is inside and then pairs of elements forming quadrilaterals are checked using a criteria and if necessary the internal diagonal is swapped. In Delaunay the criteria is maximising the minimum angle in the pair of triangles. When all the nodes have been inserted using either method then any triangles which include artificial nodes are deleted, and the resulting triangulation is Delaunay.

It should be noted that Delaunay does have a degenerate case when two (or more) elements have the same circumcircle, in this case either triangulation is Delaunay, the triangulation is not unique, (see Figure 7), and we are free to choose either diagonal.

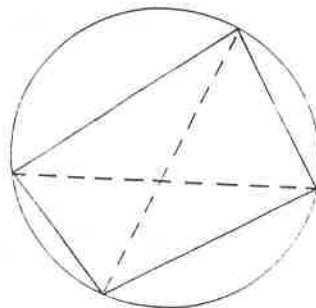


Figure 7. Degenerate case of Delaunay.

Delaunay is used with the Finite Element Method, since the equiangular property produces well-conditioned stiffness matrices (Hunt [5]).

An alternative non data dependent triangulation is

(2) Minimum Weight Triangulation (MWT)

This triangulation seeks to minimise the total length of all the edges in the triangulation, (Watson [11])

$$\text{i.e. } \text{Min} \sum_{i=1}^e \| e_i \| = \text{Min} \sum_{T_L \in T} \sum_{i,j \in T_L} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} .$$

This has the advantage of being useful for interpolation on the triangles as the nearest neighbours of a point are connected to it. However, it does not have any properties like the equiangular property of Delaunay.

(b) Data Dependent Extensions of the Previous Criteria

(3) The MAX-MIN Angle Criterion

This criterion is a data dependent extension of the Delaunay triangulation. Rather than trying to maximise the minimum angle in a set of triangles with vertices given by V , we try to maximise the minimum 3-D angle in a set of triangles with vertices given by W .

This is helped by remembering that

$$\cos (\angle_{ijk}) = \frac{(W_k - W_j, W_i - W_j)}{\|W_k - W_j\| \|W_i - W_j\|} .$$

This criteria does not work well at all.

Delaunay can be thought of as the max-min protected angle of W onto V .

(4) MWT - 3D

This is the Data Dependent extension of the MWT where we seek to minimise the sum of the 3-D lengths of the edges of all the triangles, with vertices given by W . T_L is a triangulation

$$\text{i.e. Min } \sum_{T_L \in T} \sum_{i,j \in T_L} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (F_i - F_j)^2} .$$

Hence MWT works on a protection of W onto V .

(c) Nearly C^1 Data Dependent Criteria

The following criteria are based on the premise that if the surface produced by the interpolating function is as smooth as possible then the errors in interpolating the underlying function will be reduced.

(5) Angle Between Normals (ABN)

The Angle between Normals cost function seeks to measure the angle between the normals to the planes, produced by the piecewise linear interpolant in R^3 .

Let $\underline{n}^{(1)}$ and $\underline{n}^{(2)}$ be the normal vectors to the two planes P_1 and P_2 respectively which meet at an edge i.e.

$$\underline{n}^{(i)} = \frac{1}{\sqrt{a_i^2 + b_i^2 + 1}} \begin{bmatrix} a_i \\ b_i \\ -1 \end{bmatrix} \quad i = 1, 2 .$$

The cost function is the acute angle θ between these two vectors, i.e.

$$s(f_T, e) = \theta = \cos^{-1} A.$$

$$A = \frac{a_1 a_2 + b_1 b_2 + 1}{\sqrt{a_1^2 + b_1^2 + 1} \sqrt{a_2^2 + b_2^2 + 1}}$$

i.e.

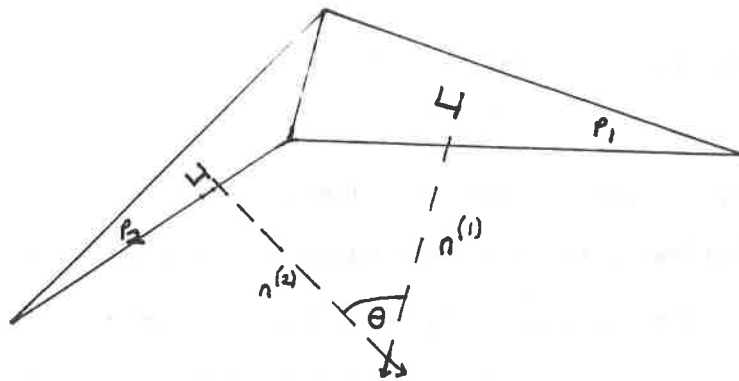


Figure 8. Angle θ as found for criteria ABN.

This criterion is the most intuitively satisfying as it should provide smoothness even near edges of the domain Ω . Another criteria based on the concept of underlying smoothness is:

(6) Jump in Normal Derivatives

This case function is a measure of the jump in the normal derivatives of P_1 and P_2 across the common edge e , i.e.

$$s(f_T, e) = |n_x(a_1 - a_2) + n_y(b_1 - b_2)|$$

where $n = \begin{bmatrix} n_x \\ n_y \end{bmatrix}$ is a unit vector in the perpendicular direction of the edge e .

$$n = \frac{m}{\|m\|} \quad \text{where} \quad m = v_k + (\lambda-1)v_i - \lambda v_j$$

$$\lambda = \frac{(v_j - v_i, v_k - v_i)}{(v_j - v_i, v_j - v_i)}$$

This however does not seem as intuitively obvious and so for smooth functions the Angle Between Normals criterion is probably better.

This brings us to the next category of criteria in which we try to get some measure of how close the planes formed in R^3 by the interpolants on neighbouring triangles are to being co-planar.

d) Near Planar Criteria

These criteria relate to cost functions defined on interior edges of the triangulation, and attempt to give some measure of how near to being planar two triangles, T_1 and T_2 , with a common edge e are.

(7) Plane-fit (PF)

The plane-fit criteria measures the error between the linear interpolants P_1 and P_2 interpolated at the other vertex V_ℓ, V_k respectively in the quadrilateral Q and the actual function value F_ℓ, F_k respectively

$$\text{i.e. } S(f_T, e) = \|h\|$$

$$\text{where } h = \begin{bmatrix} |P_1(x_\ell, y_\ell) - F_\ell| \\ |P_2(x_k, y_k) - F_k| \end{bmatrix}$$

Intuitively this gives a measure of how far from being planar the planes are, but it is probably best on smooth functions which do not have large second derivatives.

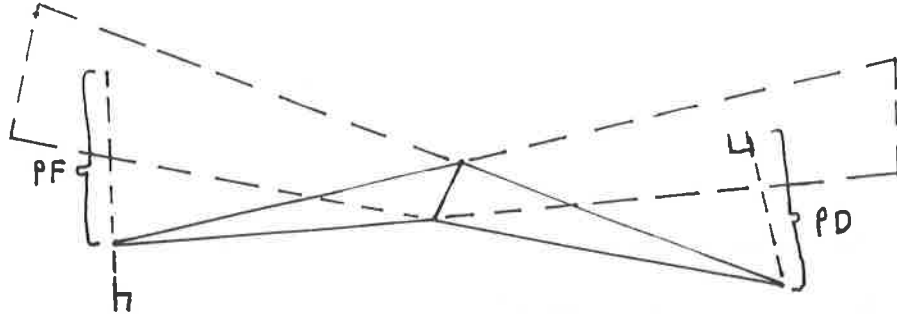


Figure 9

(8) Plane Dist (PD)

The Plane-Dist Criteria measures the distance between the extended planes P_1 and P_2 and the points w_ℓ and w_k respectively.

i.e. $S(f_T, e) = \| h \|$

$$h = \begin{bmatrix} \text{Dist} (P_1, F_\ell) \\ \text{Dist} (P_2, F_k) \end{bmatrix}$$

where $\text{Dist} (P_m, F_n) = \frac{|P_m(x_n, y_n) - F_n|}{\sqrt{a_m^2 + b_m^2 + 1}}$.

This has the same advantages and disadvantages as the plane-fit criterion.

In both near planar criteria the norms used are the norms which correspond with the vector ordering norm, with the infinity norm corresponding with the lexicographic ordering.

We now come to the final criterion.

(9) Approximate Equidistribution

This criteria seeks to minimise a cost function along the edges in the triangulation.

The criteria is based on that proposed by Sweby [10]. The underlying idea is that, in 1-D, behaviour of the underlying function u can be monitored by looking at the integral of some function w of u over an interval $[a,b]$,

$$\text{i.e. look at } \int_a^b w(u(x))dx .$$

The monitor function, w , is some function of u which allows us to look at its behaviour, i.e. u_{xx} to look at curvature. This idea can be used to position nodes to minimise errors.

Carey and Dinh [1] showed that it is possible to choose the monitor function so that the error given by a k 'th degree interpolating polynomial in the H_m semi-norm is minimised.

We look at the weighting function in 2-D, integrated along interior edges and minimised in each strictly convex quadrilateral. We will look at the L_2 -Norm as is the norm which is used in calculating errors in the Finite Element Method. Hence in 1-D

$$w = [u_{xx}]^{3/2}$$

and in 2-D, the directional analogue is,

$$w = (\cos^2\theta u_{xx} + 2\cos\theta \sin\theta u_{xy} + \sin^2\theta u_{yy})^{3/2}$$

where $\tan\theta = \frac{y_j - y_i}{x_j - x_i}$

and the cost function, S , is

$$S(f_T, e) = \int_{v_i}^{v_j} w \, ds .$$

In the test cases u_{xx} , u_{xy} and u_{yy} could all be evaluated exactly so finding $\int_{v_i}^{v_j} w \, ds$ was fairly easy using Gaussian quadrature.

This criteria could be quite useful as it would fit in with a node placement technique based on equidistribution.

These are all the criteria that we looked at, in the next section we look at the implementation details.

5. IMPLEMENTATION DETAILS

In this section we look at the different strategies that were used to implement each criteria. We look at different Local Optimization Procedures, the differences caused by searching the lists of triangles in a different order and the use of a strict or non-strict inequality is the swapping criteria.

(a) Local Optimisation Procedures (LOP)

In a truly "Local" LOP we would check just the costs across the edges $e = \overrightarrow{v_i v_j}$ and $e' = \overrightarrow{v_k v_\ell}$. (see figure 9) and then keep the edge which has the smallest cost.

In the "global" LOP used by Dyn, Rippla & Levin [3], they set up a vector cost \underline{S} so that the effect of changing from e to e' is monitored across not only the internal diagonals but also the other four edges, $\overrightarrow{v_i v_\ell}$, $\overrightarrow{v_i v_k}$, $\overrightarrow{v_\ell v_j}$ and $\overrightarrow{v_j v_k}$, (see figure 10), as well. The resulting vectors \underline{S} and \underline{S}' are then compared using the specified vector Norm.

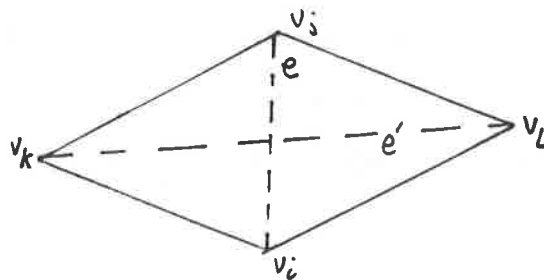


Figure 10. Possible triangulation of a convex quadrilateral.

It should be noted that MWT, MWT-3D, MAX-MIN angle and equidistribution are already "Local" LOP, and that PF and PD rely on a specified norm since the cost function across an edge is a vector.

(b) Differences in triangle searching

In searching to diagonals to swap, each triangle in the list of triangles is taken in turn and neighbouring elements sought (i.e. those with a common edge), in the rest of the list. On finding a neighbouring element the two are considered as forming a quadrilateral in which, if appropriate, the diagonal can be swapped. If no swapping occurs, another neighbour is sought, or if all neighbours have been found then the next triangle is taken as a base. However if swapping does occur there are two options: either to take one of the newly formed triangles as base and continue searching for its neighbours or to jump to the next triangle in the list.

The consequences can be that totally different grids are produced with almost identical costs.

(c) Swapping strategies

As described in Section 2 we can define an ordering on the set of triangulation, and form the definition that

$$T \leq T' \quad \text{if} \quad \underline{S} \leq \underline{S}' \quad \text{according to some vector ordering.}$$

From this definition we can produce two strategies to change the internal diagonals, e and e' with cost vectors \underline{S} and \underline{S}' respectively.

A sweep is a full sweep through the list of triangles.

The strategies are:-

- i) Change from edge e to edge e' if \underline{S} is less than or equal to \underline{S}' in the ordering, storing the number of swaps on each sweep and the number of exact equalities on each sweep, when all the swaps in a sweep are due to equality then change the strategy and change only if \underline{S} is strictly less than \underline{S}' , and keep this strategy until there are no swaps in a sweep when the process is terminated.
- ii) Change from edge e to edge e' if \underline{S} is strictly less than \underline{S}' in the ordering and continue until there are no swaps in a sweep. Then do one sweep where edge e is changed to edge e' if \underline{S} is less than or equal to \underline{S}' . Then continue with the original strict inequality until there are no more swaps on one sweep.

The consequences of such strategies are discussed in a further section while in the next section we outline the test functions and data sets used.

6. TESTING PROCEDURE

In this section the underlying functions and the sets of data points, which were used in the calculations are detailed and the procedure for error comparison is outlined.

1) Data Sets

Three different data sets were used in the calculations. The first two data sets were those of 33 and 100 points which were used by Franke [4] and Dyn, Rippla and Levin [3]. The third set was a set of 81 points set on a regular cartesian grid. The 33 point and 100 point data sets are presented in Figures 11(a) and 11(b) respectively.

2) Test functions

The test functions were mainly smooth curved surfaces, although function F7 had discontinuous first derivatives, and the numbering is that used by Dyn, Rippla and Levin [3].

Only the functions detailed here showed any real improvement in representations so detailed expressions for these are given. All the functions are defined on the unit square

$$F1 = .75 \exp \left[- \frac{(9x-2)^2 + (9y-2)^2}{4} \right] + .75 \exp \left[- \frac{(9x+1)^2}{49} - \frac{(9y+1)^2}{10} \right] \\ + .5 \exp \left[- \frac{(9x-7)^2 + (9y-3)^2}{4} \right] - .2 \exp \left[- (9x-e)^2 - (9y-7)^2 \right]$$

$$F2 = \frac{\tanh(9y-9x) + 1}{9}$$

$$F7 = \begin{cases} 1 & \text{if } y - \xi \geq \frac{1}{2} \\ 2(y - \xi) & \text{if } 0 \leq y - \xi \leq \frac{1}{2} \\ \frac{\cos(4\pi r) + 1}{2} & \text{if } r \leq \frac{1}{4} \\ 0 & \text{otherwise} \end{cases}$$

where $\xi = 2.1x - 0.1$

$$r = \sqrt{(\xi - \frac{1}{2})^2 + (y - \frac{1}{2})^2}$$

$$F8 = \tanh(-3g(x,y)+1) \quad .g(x,y) = 0.595576(y + 3.79762)^2 - x - 10$$

$$F9 = \left[1 - \frac{x}{2}\right]^6 \left[1 - \frac{y}{2}\right]^6 + 1000(1-x^4)^3 x^3 (1-y)^3 y^3 + y^6 \left[1 - \frac{x}{2}\right]^6 + x^6 \left[1 - \frac{y}{2}\right]^6$$

For pictures of all functions except F8 see figures 13 and 14.

3) Error Calculations

On each triangulation the linear interpolating function, f_T , was constructed and the error between f_T and the actual function F was computed on a grid of 33×33 nodes, uniformly placed over the unit square. The mean pointwise error, the RMS error and the maximum error were calculated. The maximum error was not instructive as it tended to occur on the convex hull of the unit square and was usually due to the sparsity of points on the convex hull, so the figures used in the tables are the mean pointwise errors.

7. RESULTS

In this section we present numerical and graphical results which show how good piecewise linear interpolations of data can be achieved by using data dependent criteria to generate the underlying grids. The effects of the differing procedures as detailed in Section 5 are also discussed.

The numerical results presented here are the main pointwise error for the different functions and criteria, found as detailed in Section 6.

The results for the 81 point data set are not reproduced here as there were only slight changes in numerical and graphical results. The graphical results show 3-D isoparametric surface representations (isoplots), and contours produced by the interpolants on the grids, which are trying to represent the actual functions which are presented in this form in figures 13 and 14. Various grids which are produced are also shown in figures 17 and 18. The data sets used, 33 data points and 100 data points, are presented in figures 11(a) and (b) respectively while the Delaunay triangulations of these data sets are respectively produced in figures 12(a) and 12(b).

As can be seen from the numerical results, the best criteria for all functions with 33 data points is the ABN-2 criteria and this works especially well with functions F2 and F8. ABN-2 also works well with 100 data points and improves all the functions, although the equidistribution criteria works especially well with 100 data points except on F7 which is not surprising. Equidistribution does not work quite as well with 33 data points and with function F7 it actually increases the error although this is due to the fact that much of the function, i.e. the ramp and both planes have constant slope and so only the mountain actually has non-zero 2nd derivatives and on the mountain the sparsity of points means that the criteria cannot work well.

POINTWISE ERRORS FOR 33 POINTS

METHOD	FUNC 1	FUNC 2	FUNC 7	FUNC 8	FUNC 9
DELAUNAY	0.04886	0.01604	0.06866	0.15210	0.05042
MWT-3D	0.05813	0.01585	0.06019	0.09926	0.04826
ABN-1	0.04033	0.00619	0.04210	0.13123	0.04633
ABN-2	0.03964	0.00610	0.04096	0.07723	0.04702
JND-1	0.04033	0.01176	0.04562	0.10457	0.04639
JND-2	0.04532	0.01107	0.04922	0.10456	0.04635
PF-1	0.04285	0.00618	0.04342	0.14730	0.04650
PF-2	0.04661	0.01190	0.05737	0.15113	0.04683
PD-1	0.05255	0.00618	0.04267	0.11505	0.04737
PD-2	0.05379	0.01188	0.04790	0.11513	0.04750
EQUID	0.04541	0.00614	0.06601	0.10472	0.04697

Table 1

POINTWISE ERRORS FOR 100 POINTS

METHOD	FUNC 1	FUNC 2	FUNC 7	FUNC 8	FUNC 9
DELAUNAY	0.01637	0.00419	0.02707	0.04033	0.01492
MWT-3D	0.02002	0.00366	0.02869	0.01884	0.01249
ABN-1	0.01546	0.00145	0.02027	0.01390	0.01101
ABN-2	0.01550	0.00147	0.01727	0.02839	0.01075
JND-1	0.01571	0.00145	0.02207	0.02961	0.01097
JND-2	0.01533	0.00151	0.02099	0.03034	0.01079
PF-1	0.01563	0.00145	0.01881	0.03011	0.01137
PF-2	0.01634	0.00182	0.02000	0.03162	0.01115
PD-1	0.01600	0.00145	0.02374	0.02557	0.01113
PD-2	0.01670	0.00152	0.02576	0.02505	0.01079
EQUID	0.02070	0.00358	0.02675	0.03805	0.01639

Table 2

The graphical results show many interesting features. Figure 15(a) shows that for function F2 with 33 points, Delaunay produces a very poor representation of the function, but figure 15(b) shows that ABN-2 can improve this tremendously. Figure 15(c) and figures 16(a) and (b) show that JND-2, PF2 and PD2 can improve the representation slightly and figure 16(c) shows that equidistribution works well in improving the representation of F2. Figures 17(a) and 18(a) show the Delaunay triangulation of 33 data points and 100 data points and figures 17(b) and (c) and 18(b) and (c) show that with the data dependent criteria, ABN-2 and equidistribution, the grids produced have many long thin triangles which run parallel to the contours.

Figure 19(a) shows the representation of F7 by a Delaunay Triangulation of 33 points. It can be seen that the "ramp" contours are not straight and that the "mountain" sprawls outwards. Figure 19(b) shows that the ABN2 criteria has produced a smoothing of the "ramp" contours and a compression of the "mountain" contours. Figure 19(c) shows that equidistribution has struggled due to the functions constant gradient sections and the sparsity of points. The "ramp" is exactly as it was with Delaunay and due to the sparsity of points and the positions of the gauss points for approximate integration the "mountain" has almost been split in two.

In figure 20(b) we can see that with 100 data points on F7 that even though the "ramp" is still unchanged from Delaunay, figure 20(a) the "mountain" is much better represented. As with 33 data points ABN-2, figure 20(b), much better represents the "ramp", but it only slightly improves the representation of the "mountain".

Differing implementations produce in all cases almost exactly the same errors but different grids, see figures 21(a) and (b) which were produced by differing the order of taking triangles when a change had occurred. A likely explanation for this phenomenon is that we are just finding local minima rather than global minima. Possible modifications to the search procedure are discussed in the next Section.

8. SUMMARY

There is an increasing use of numerical techniques which require an initial triangulation able to well represent the initial data when this is interpolated on it. It is necessary therefore to develop methods of constructing triangulations according to data dependent criteria in order to achieve this aim. It is intuitively clear that such methods exist and in this report we have detailed some that might be used.

Using a Delaunay triangulation as a starting grid we have applied a local optimisation procedure to obtain a locally optimal triangulation with respect to a given data dependent criteria. As can be seen from the graphical results, the resulting triangulations are a lot smoother in representing the underlying data function than the original Delaunay grid.

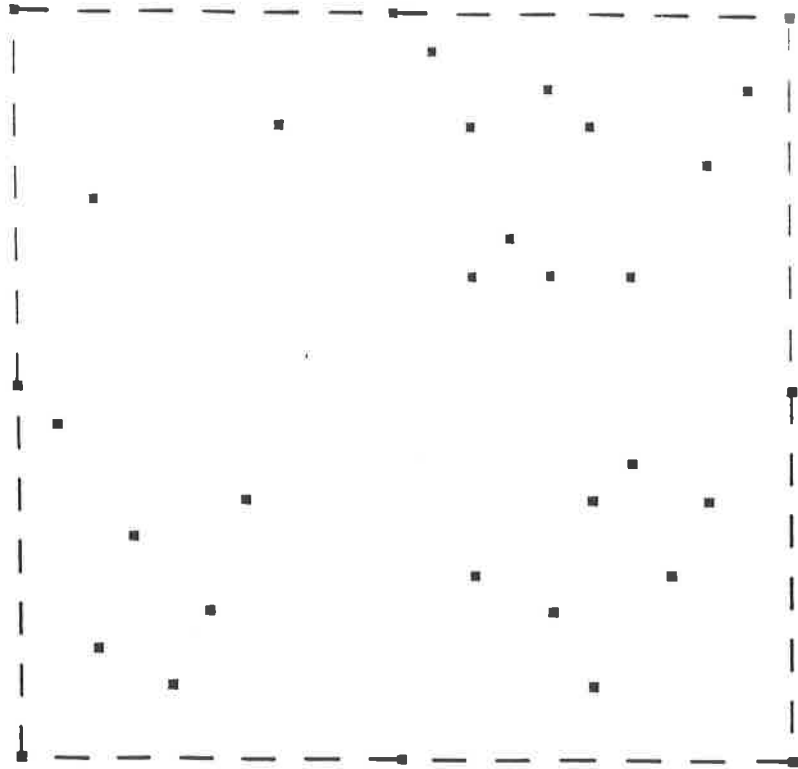
A disadvantage of the local optimisation procedure, however, is that it is not guaranteed to find the global minimum of the cost function but it is likely to find first a local minimum. A possible improvement could be made by modifying the implementation, for example, checking the edges with the largest costs first rather than going through the list sequentially. Alternatively, not using an initial triangulation but instead using diagonal swapping when inserting nodes to construct an optimal grid at each stage of node insertion. This is an area for further research.

Of the methods surveyed here, the best schemes seem to be those based on the nearly C^1 criteria, e.g. the angles between normal criteria. In almost every case these criteria reduced the error produced by the starting grid and in a number of cases significantly reduced the error. The equidistribution based criteria produced better results than the starting grid in most cases, and for 100 data points was usually as good as the nearly C^1 criteria.

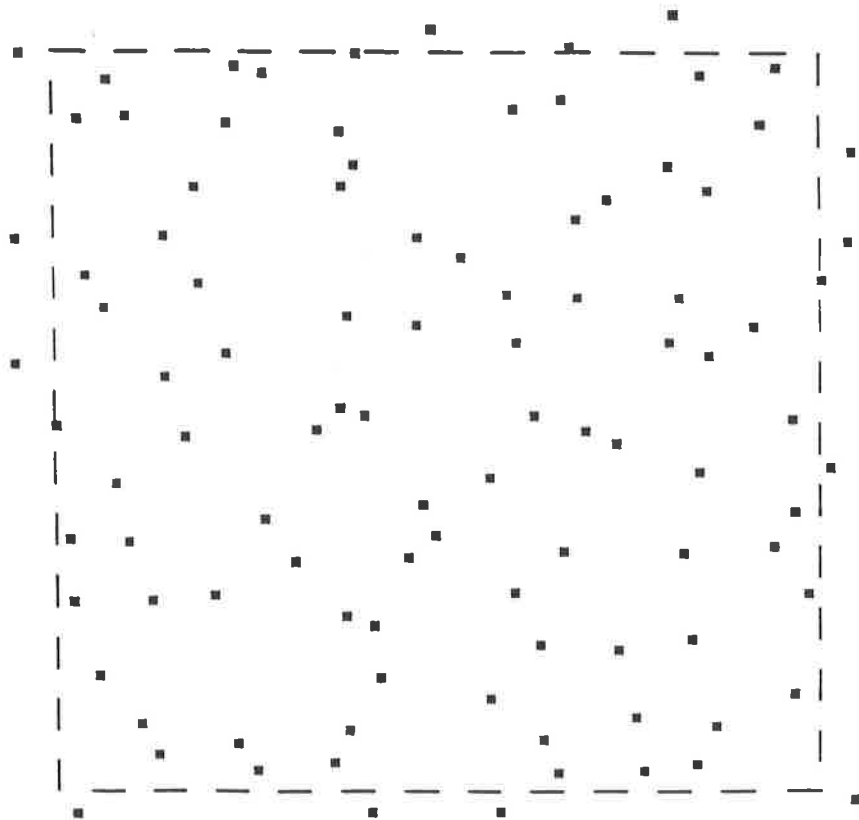
Future work will involve moving the data points, via some criteria, while keeping the connectivity of the data points the same, and eventually, combining node reconnection and node repositioning to produce better data representations. It is hoped that some of the criteria surveyed here will be useful in the future work.

REFERENCES

- [1] Carey & Dinh: *Grading Functions & Mesh Redistribution*. (1985) SIAM J. Numer. Anal., Vol. 22, pp. 1028-1040.
- [2] Delaunay: *Sur la sphere vide* (1934), Bull. Acad. Sci. USSR (VII),
- [3] Dyn, Rippa & Levin: *Data dependent triangulations for piecewise linear interpolation*, (1990) IMA J. Num. Anal. 10, pp. 137-154.
- [4] R. Franke: *A critical comparison of some methods for interpolation of scattered data*, Report NPS-53-79-003, Naval Postgraduate School, 1979.
- [5] C. Hunt: *Finite Element Meshes using Delaunay Triangles*, M.Sc. dissertation, Reading University, 1988.
- [6] C.L. Lawson: "Software for C^1 interpolation" in *Mathematical Software III*, J.R. Rice (ed.), Academic Press, New York, 1977, pp. 161-194.
- [7] K. Miller & R. Miller: *Moving finite elements*, Part 1, SIAM J. Num. Anal. 18 (1981) pp. 1019-1032.
- [8] Mock: *Tetrahedral elements and the Scharfetter-Gummel Method*, Proc. NASECODE IV (1985).
- [9] Sibson: *Locally equiangular triangulations* (1978), COMP J21. pp. 243-245.
- [10] P.K. Sweby: *Data dependent grids*, Numerical Analysis Report 7/87, Reading University.
- [11] Watson D.F. & Philip G.M: *Survey - Systematic triangulation*, Comp. Vision, Graph and Inge. Proc. 26 (1984) pp. 217-223.

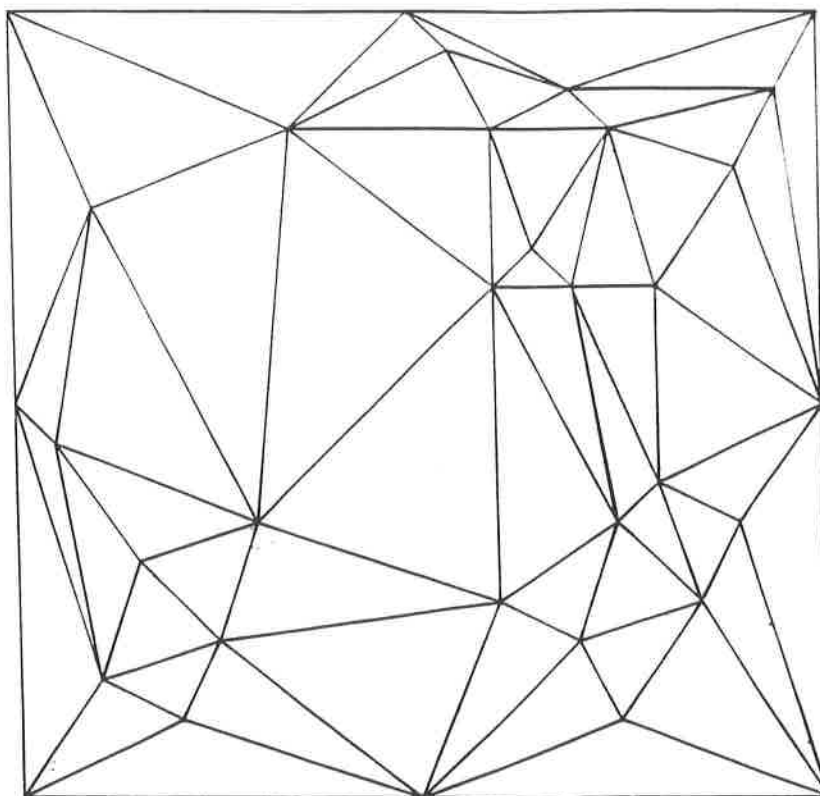


(a) POSITION OF POINTS RELATIVE TO THE UNIT SQUARE
33 DATA POINTS

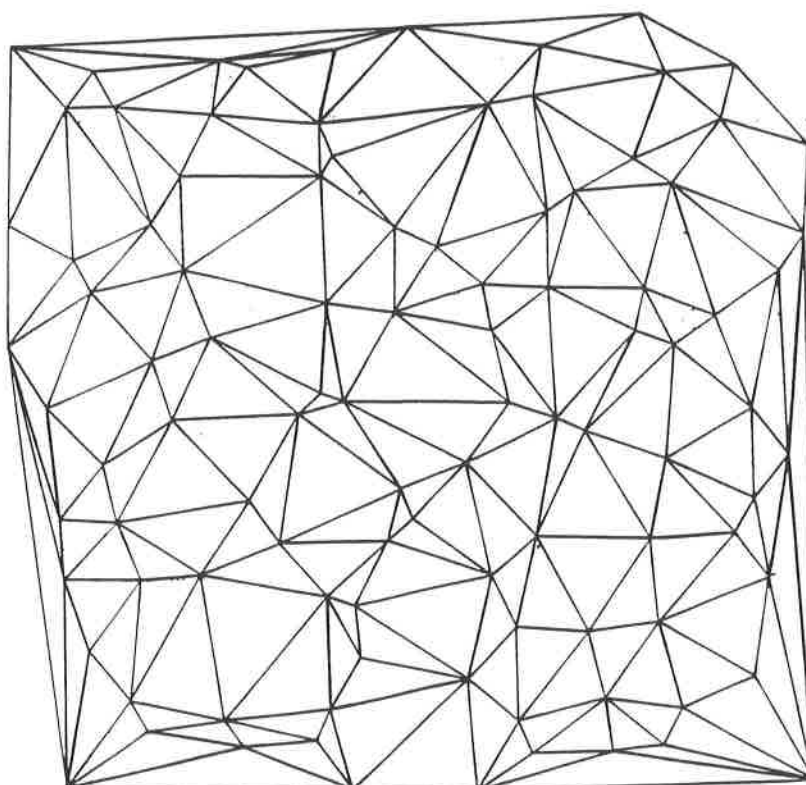


(b) POSITION OF POINTS RELATIVE TO THE UNIT SQUARE
100 DATA POINTS

Figure 11

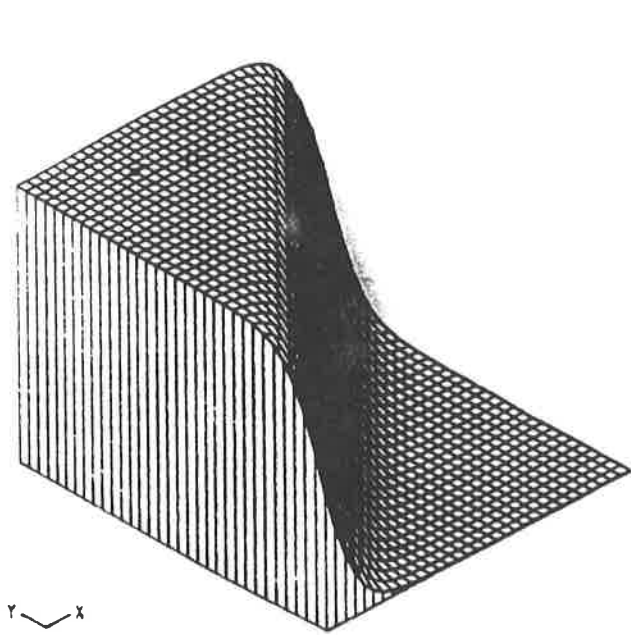


(a) DELANY TRIANGULATION WITH 53 DATA POINTS



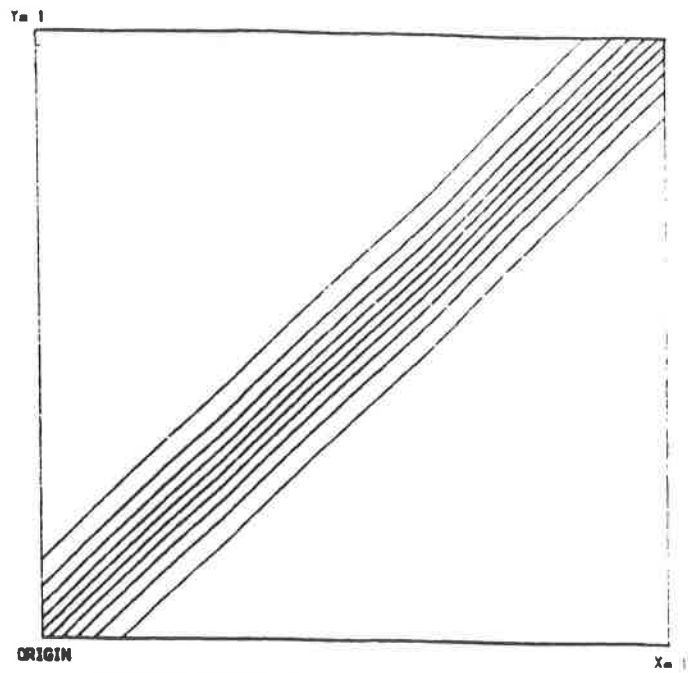
(b) DELANY TRIANGULATION WITH 100 DATA POINTS

Figure 12



Y X

ISOPLOT OF FUNCTION 2

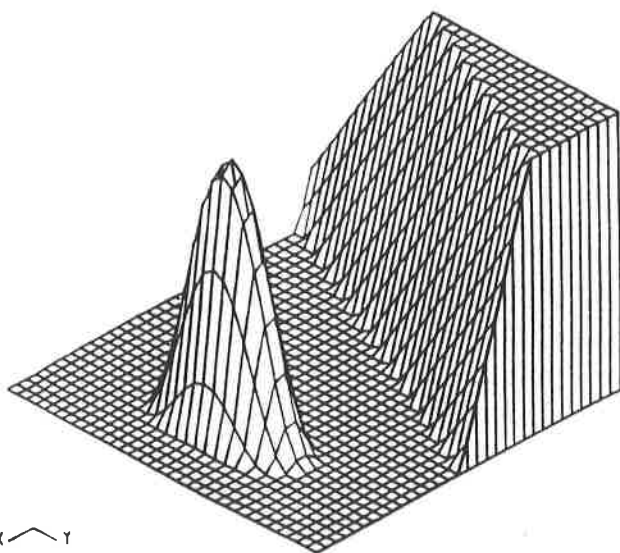


ORIGIN

X=1

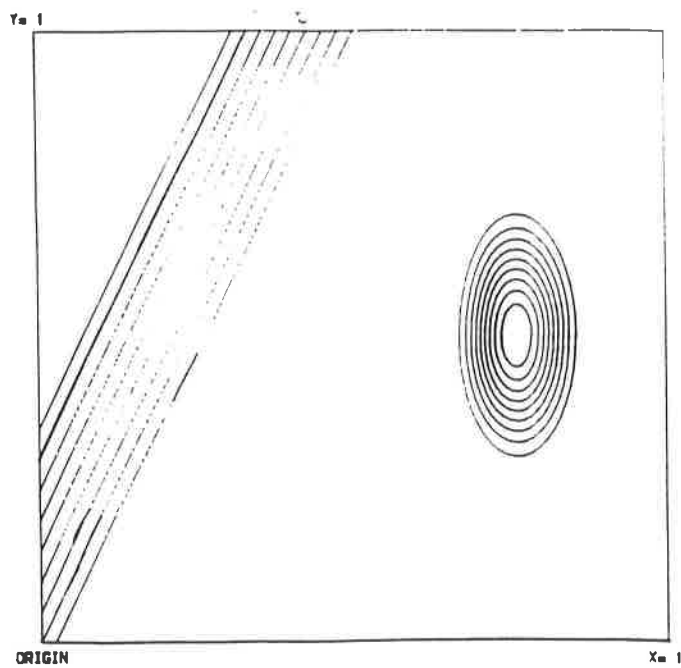
CONTOURS OF FUNCTION 2

(a)



X Y

ISOPLOT OF FUNCTION 7



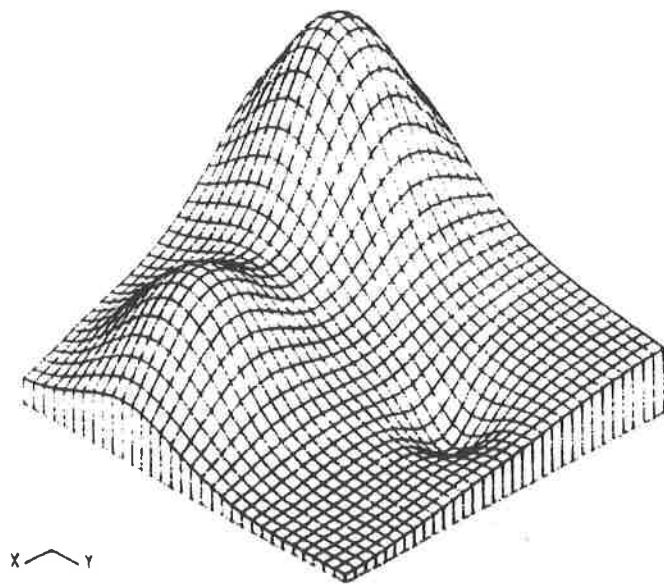
ORIGIN

X=1

CONTOURS OF FUNCTION 7

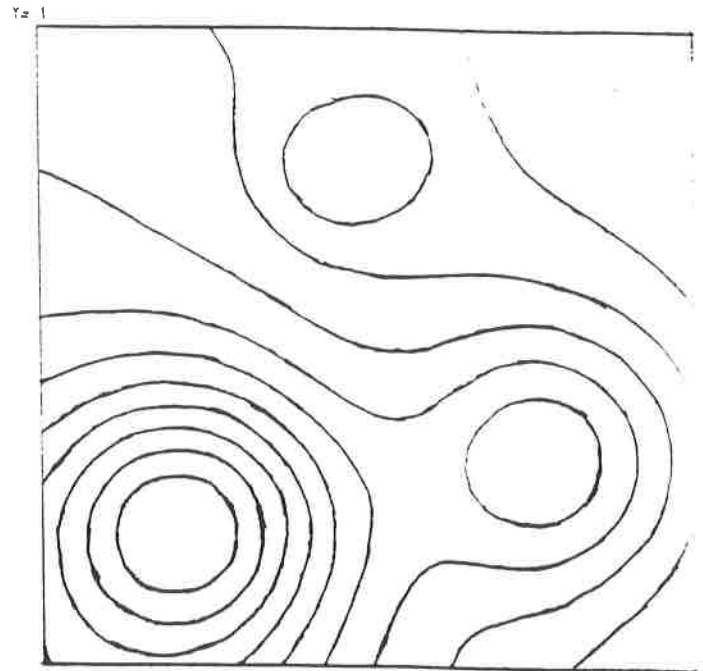
(b)

Figure 13



X Y

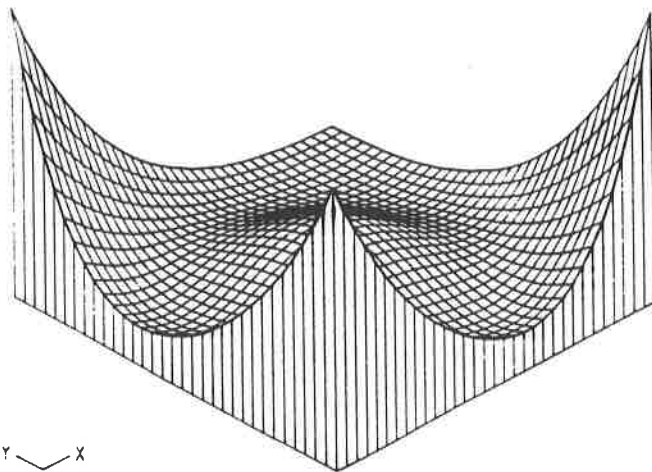
ISOPLLOT OF FUNCTION 1



ORIGIN

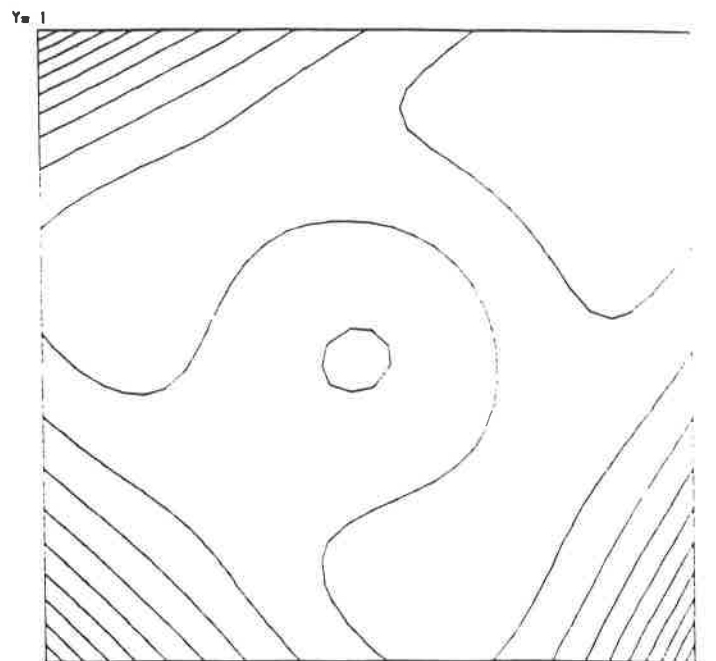
CONTOURS OF FUNCTION 1

(a)



Y X

ISOPLLOT OF FUNCTION 9



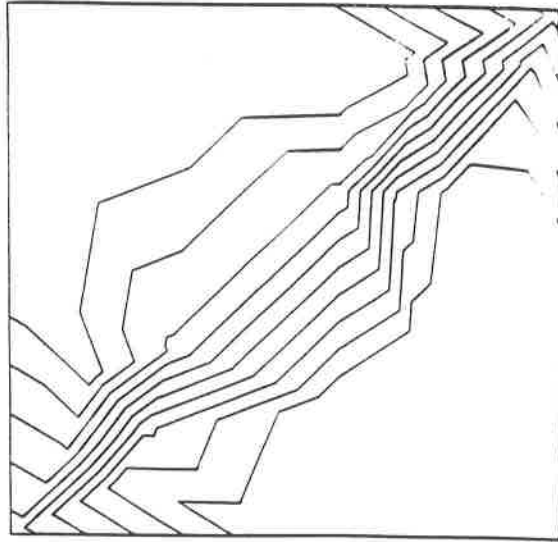
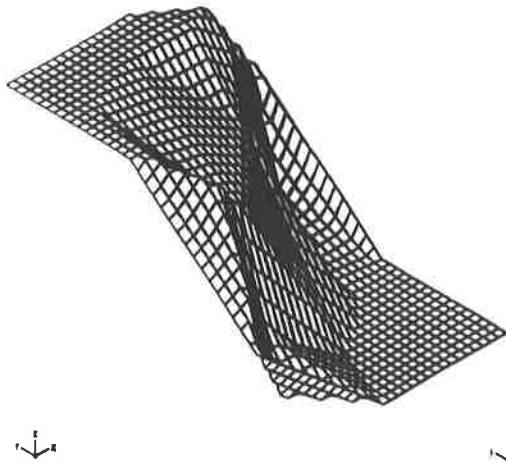
ORIGIN

CONTOURS OF FUNCTION 9

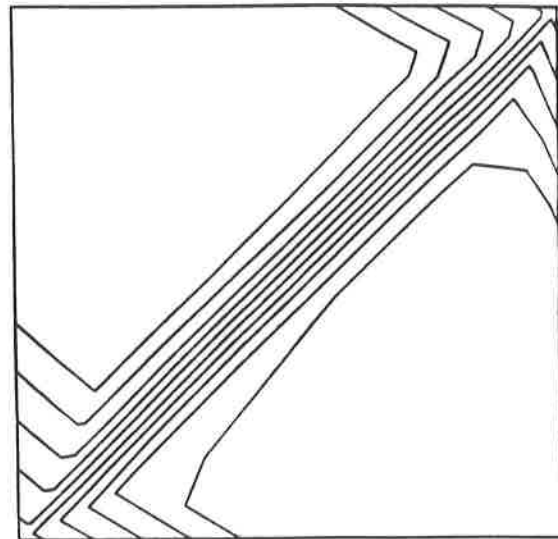
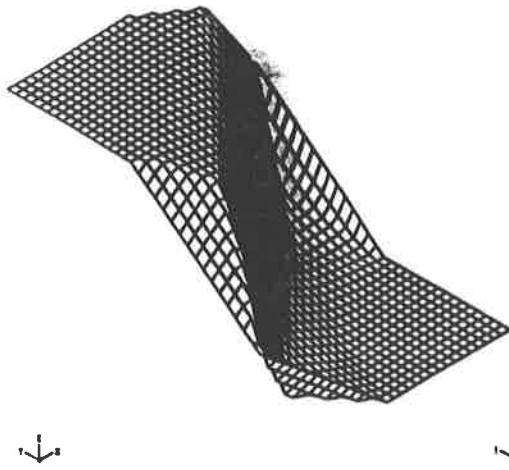
X=1

(b)

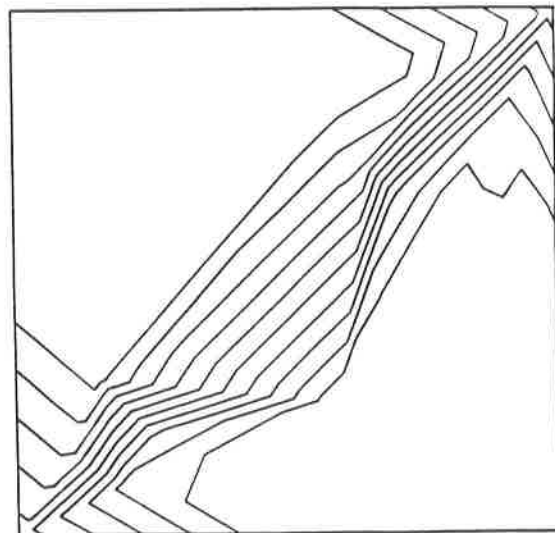
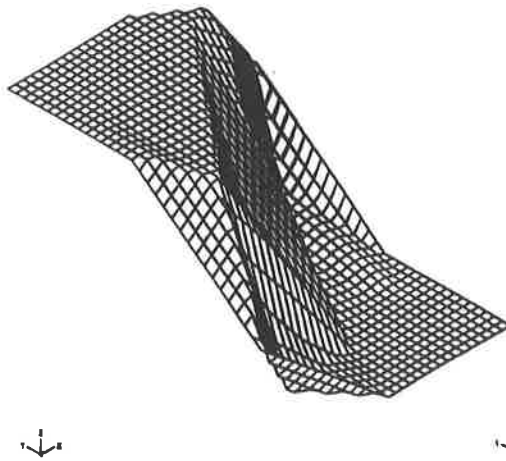
Figure 14



(a) ISOPLOT AND CONTOURS OF FUNCTION F 2 OBTAINED USING A DELAUNAY TRIANGULATION OF 33 DATA POINTS

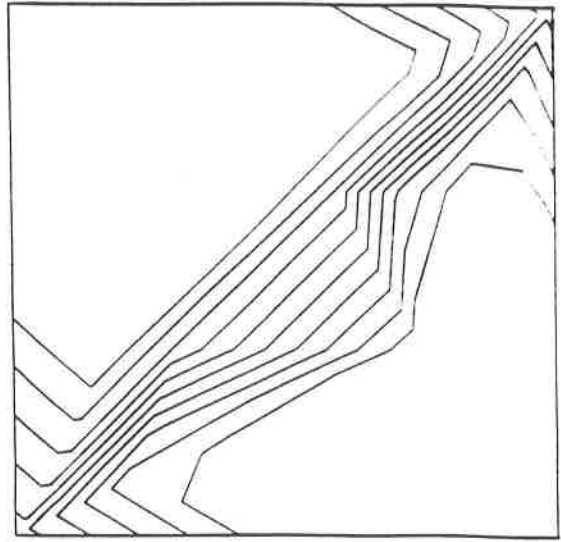
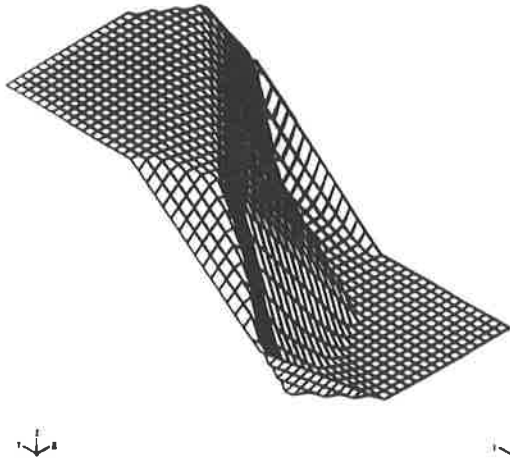


(b) ISOPLOT AND CONTOURS OF FUNCTION F 2 OBTAINED BY USING METHOD AD- 2 ON A SET OF 33 DATA POINTS

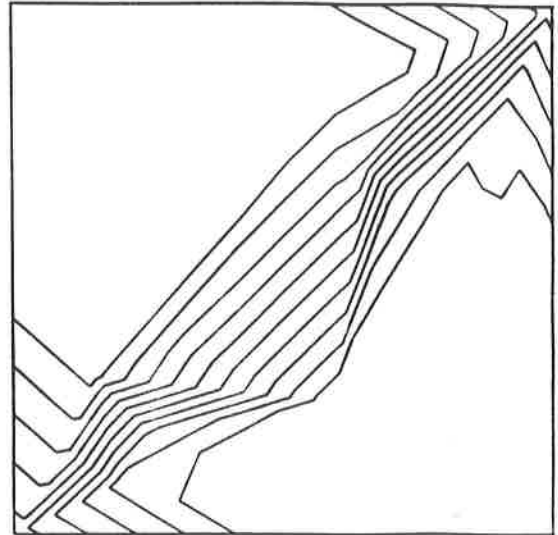
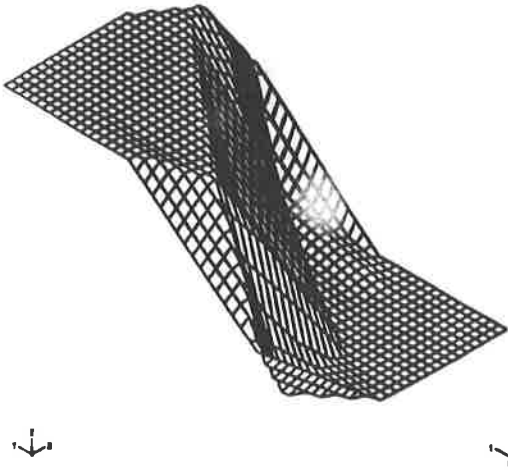


(c) ISOPLOT AND CONTOURS OF FUNCTION F 2 OBTAINED BY USING METHOD PD- 2 ON A SET OF 33 DATA POINTS

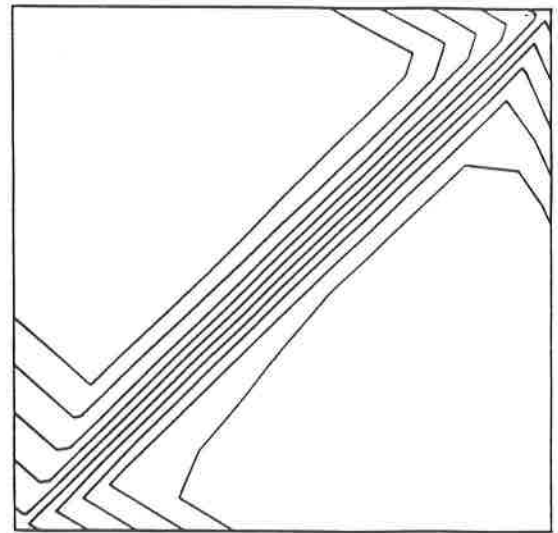
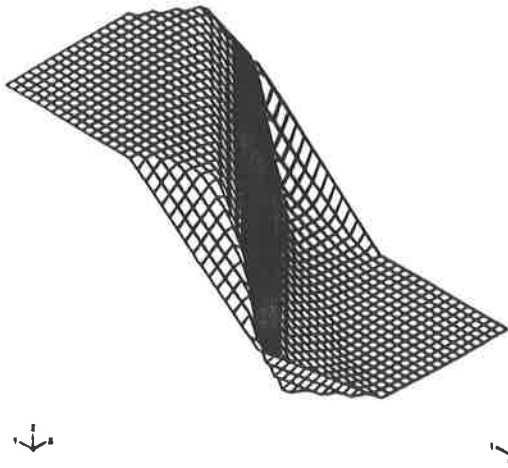
Figure 15



(a) ISOPLLOT AND CONTOURS OF FUNCTION F 2 OBTAINED BY USING METHOD JND- 2 ON A SET OF 33 DATA POINTS

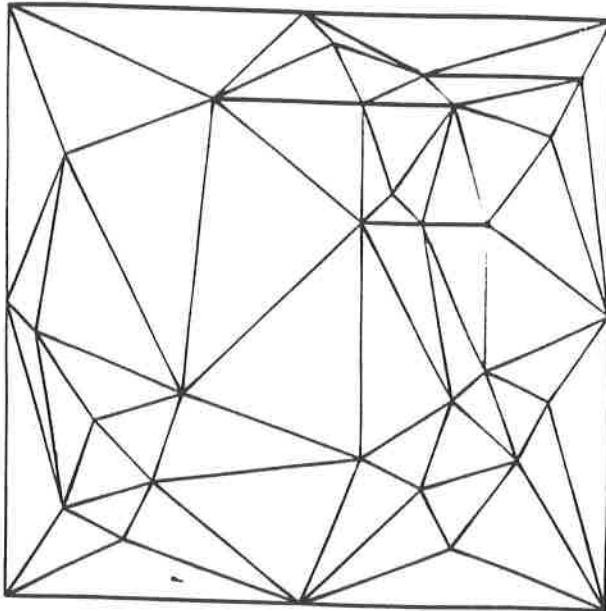


(b) ISOPLLOT AND CONTOURS OF FUNCTION F 2 OBTAINED BY USING METHOD PF- 2 ON A SET OF 33 DATA POINTS

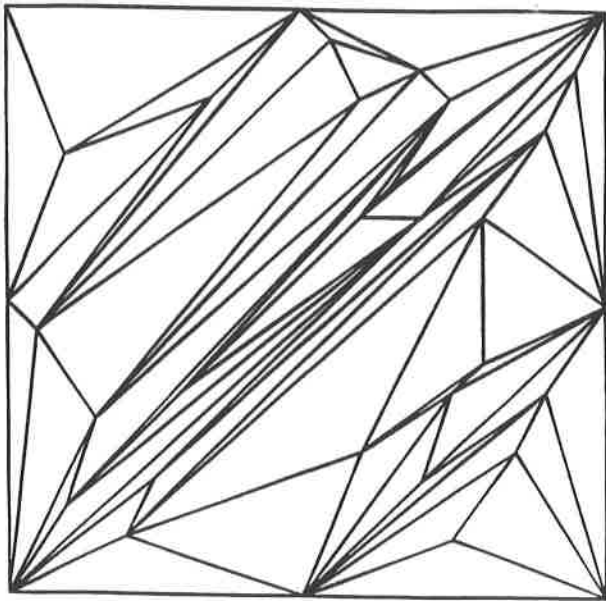


(c) ISOPLLOT AND CONTOURS OF FUNCTION F 2 OBTAINED BY USING THE EQUIDISTRIBUTION CRITERIA ON A SET OF 33 DATA POINTS

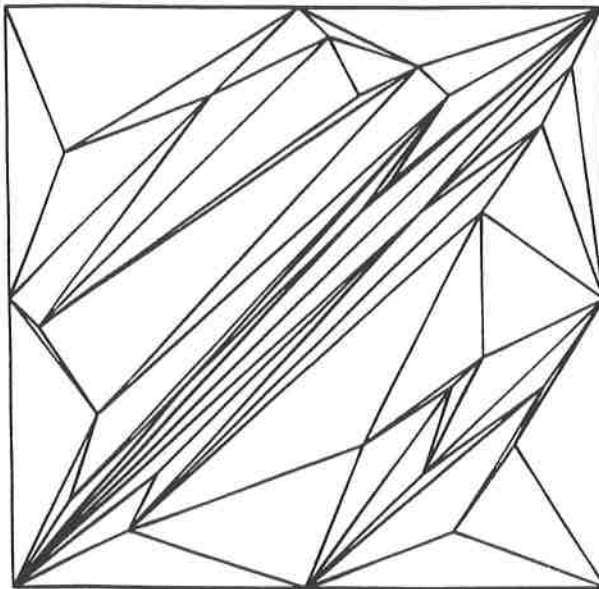
Figure 16



(a) GRID FOR DELAUNAY.FUNCTION F 2

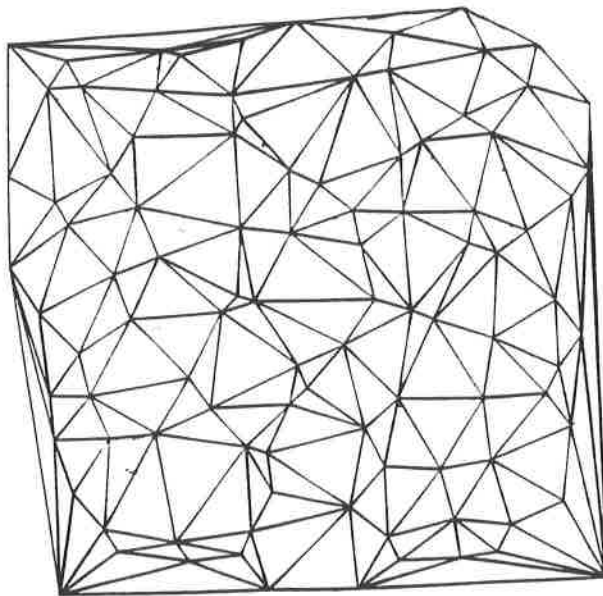


(b) GRID FOR METHOD ADM-2 .FUNCTION F 2

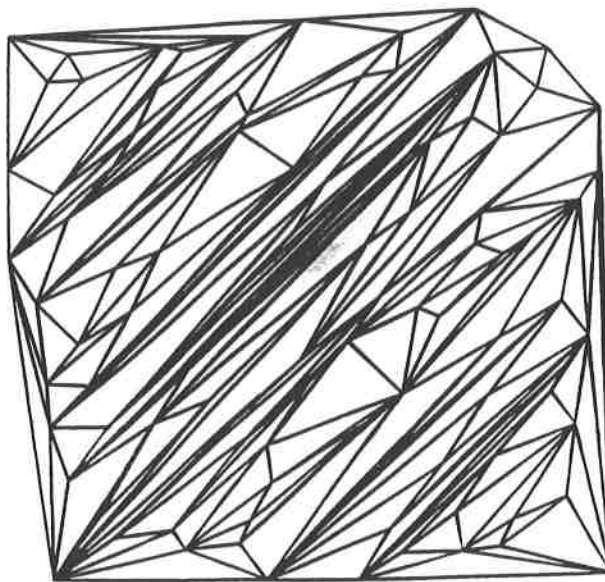


(c) GRID FOR EQUIDISTRIBUTION CRITERIA.FUNCTION F 2

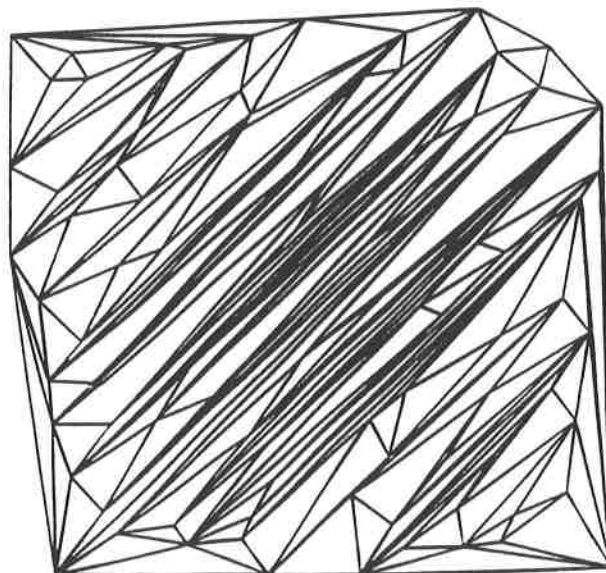
Figure 17



(a) MESH FOR SIMULATION WITH 100 DATA POINTS

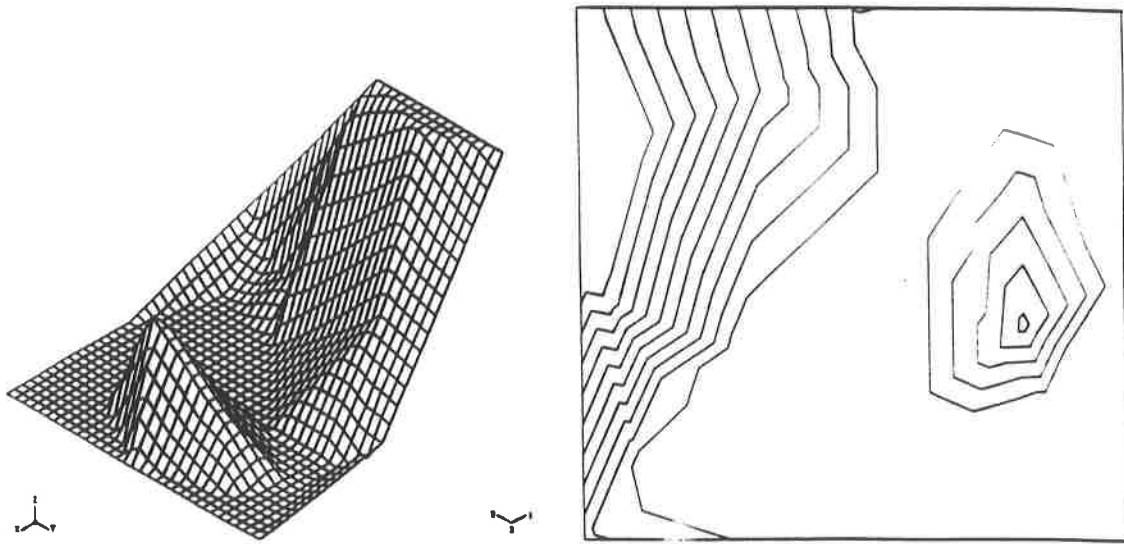


(b) GRID FOR METHOD AIN- 3 .FUNCTION F 2

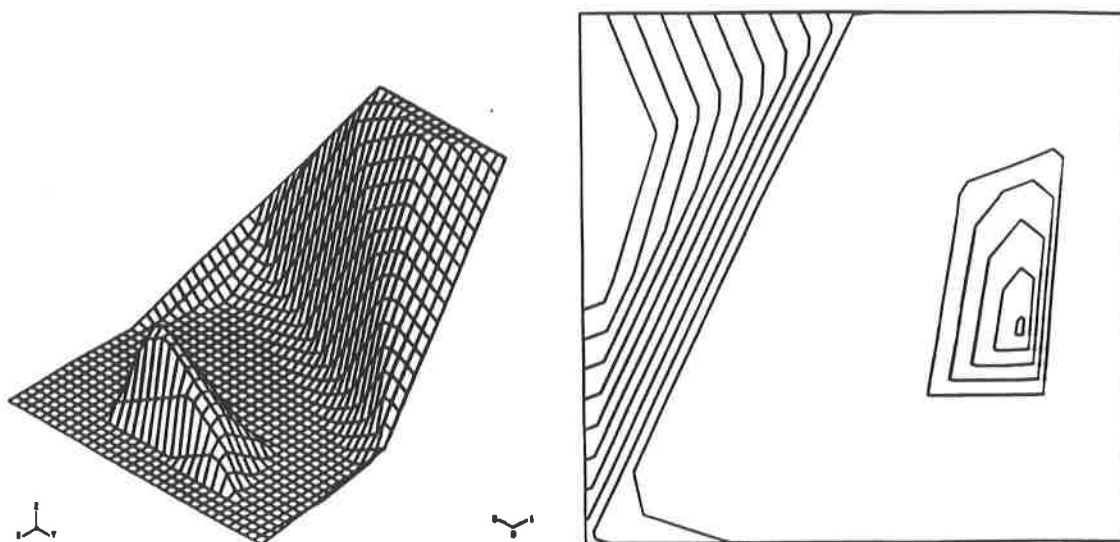


(c) GRID FOR CONCENTRATION CRITERIA.FUNCTION F 2

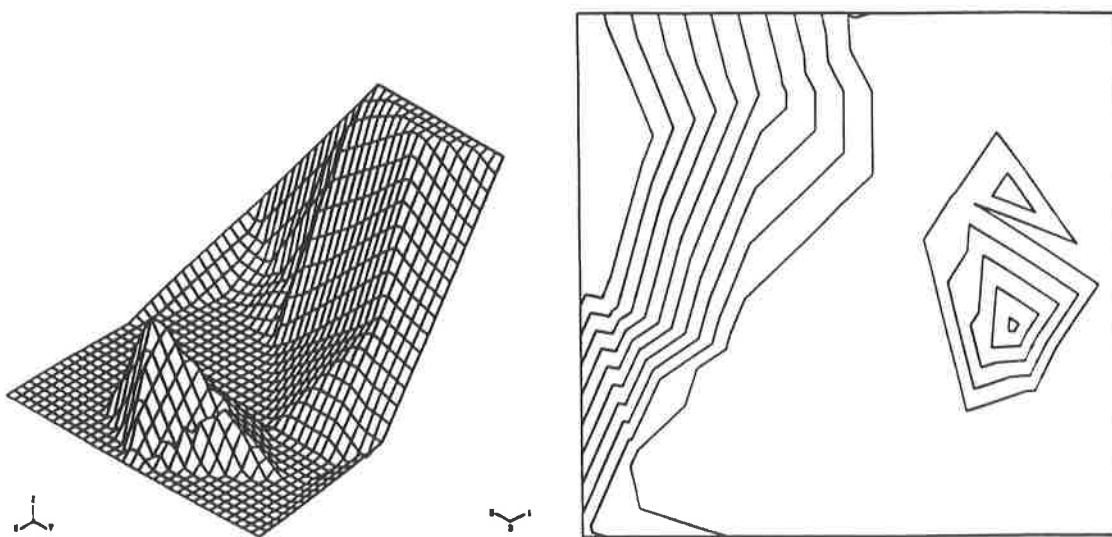
Figure 18



(a) ISOPLLOT AND CONTOURS OF FUNCTION F 7 OBTAINED USING A DELAUNAY TRIANGULATION OF 33 DATA POINTS

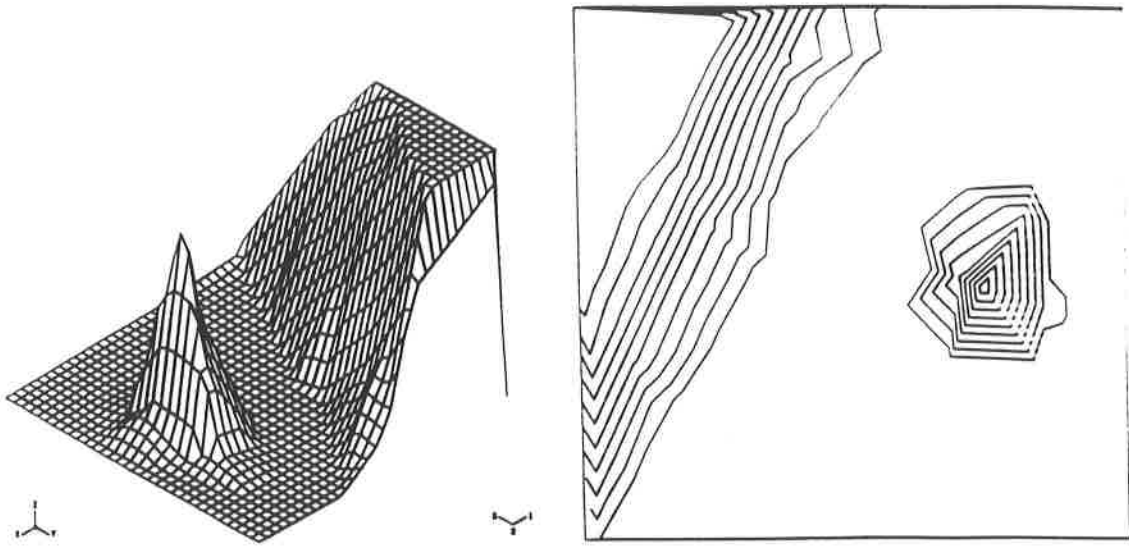


(b) ISOPLLOT AND CONTOURS OF FUNCTION F 7 OBTAINED BY USING METHOD ABM- 2 ON A SET OF 33 DATA POINTS

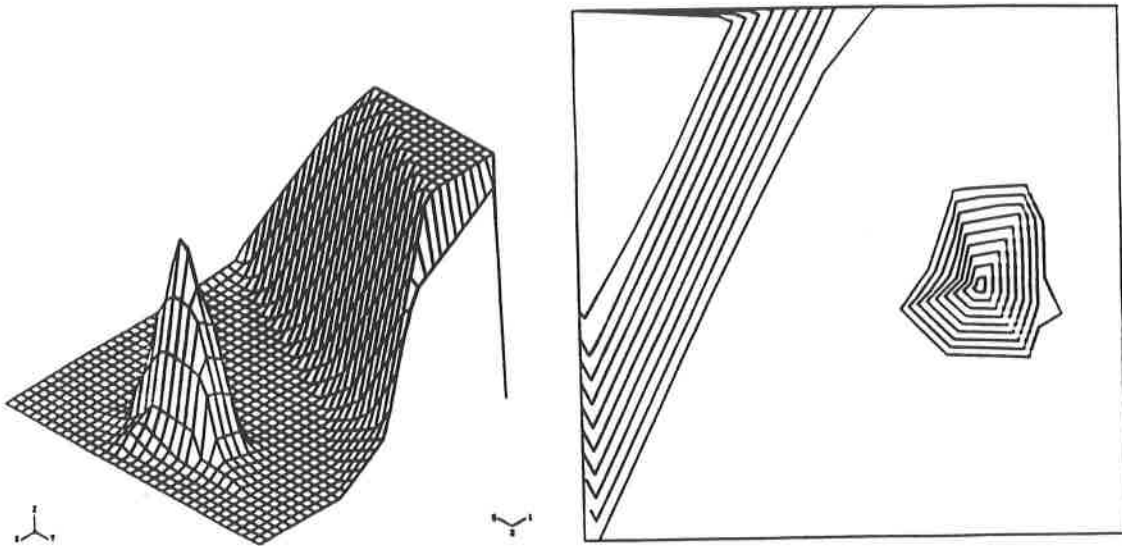


(c) ISOPLLOT AND CONTOURS OF FUNCTION F 7 OBTAINED BY USING THE EQUIDISTRIBUTION CRITERIA ON A SET OF 33 DATA POINTS

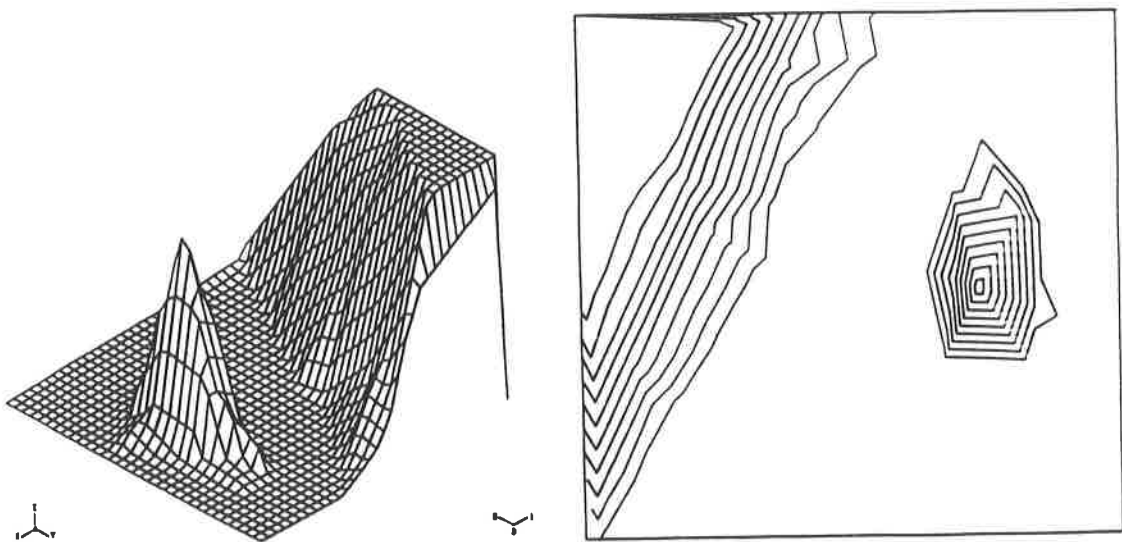
Figure 19



(a) ISOPLOT AND CONTOURS OF FUNCTION F 7 OBTAINED USING A DELAUNAY TRIANGULATION OF 100 DATA POINTS



(b) ISOPLOT AND CONTOURS OF FUNCTION F 7 OBTAINED BY USING METHOD ASM- 2 ON A SET OF 100 DATA POINTS



(c) ISOPLOT AND CONTOURS OF FUNCTION F 7 OBTAINED BY USING THE EQUIDISTRIBUTION CRITERIA ON A SET OF 100 DATA POINTS

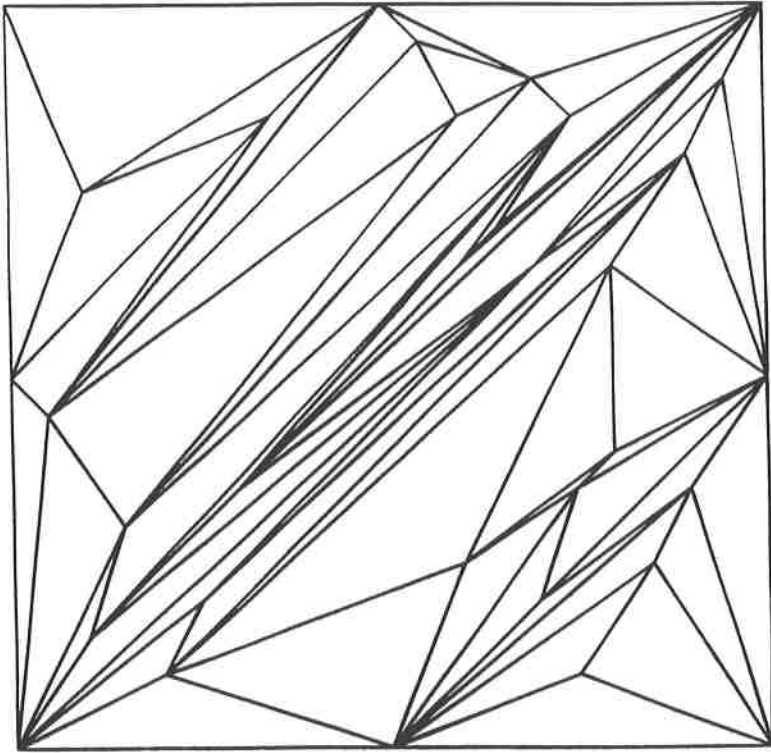


Figure 21 (a)

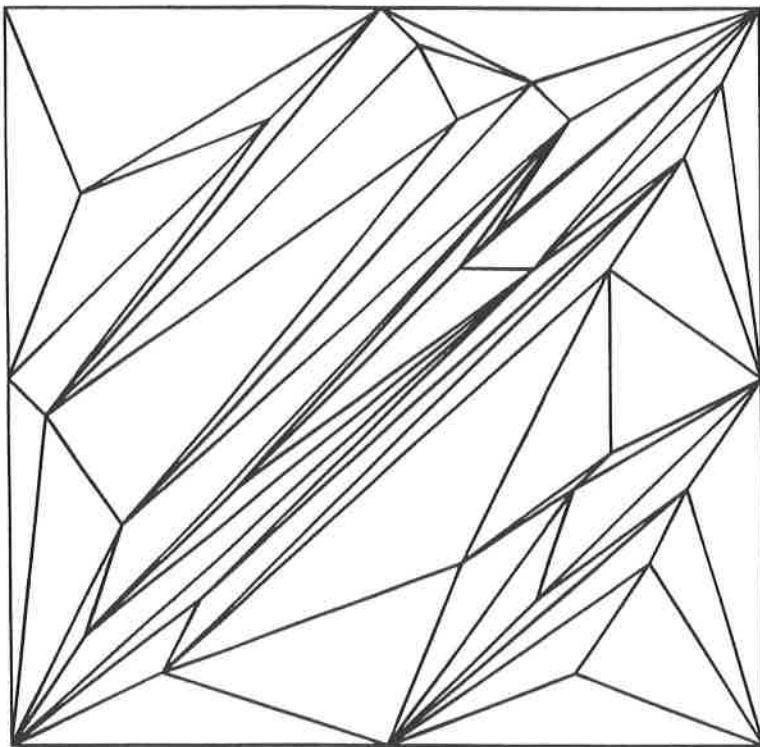


Figure 21 (b)