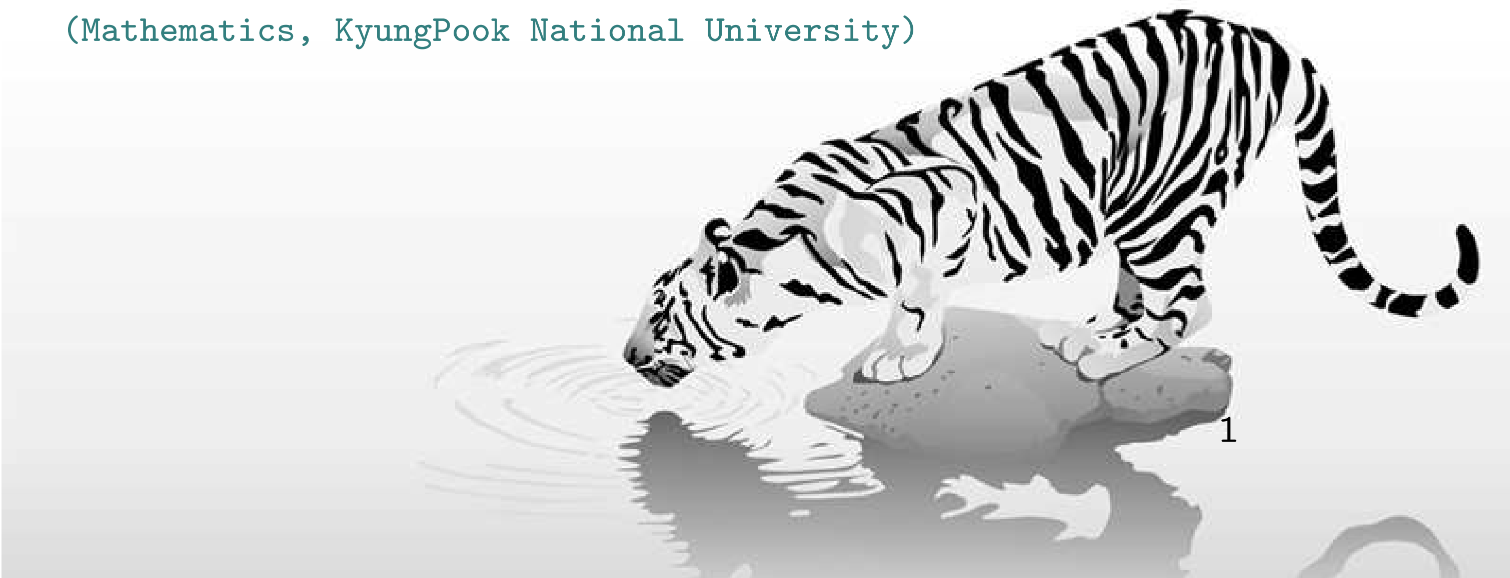
A natural heuristic method for surface reconstruction

Oct. 21, 2005

Philsu Kim
(Mathematics, KyungPook National University)



Setting of Problem

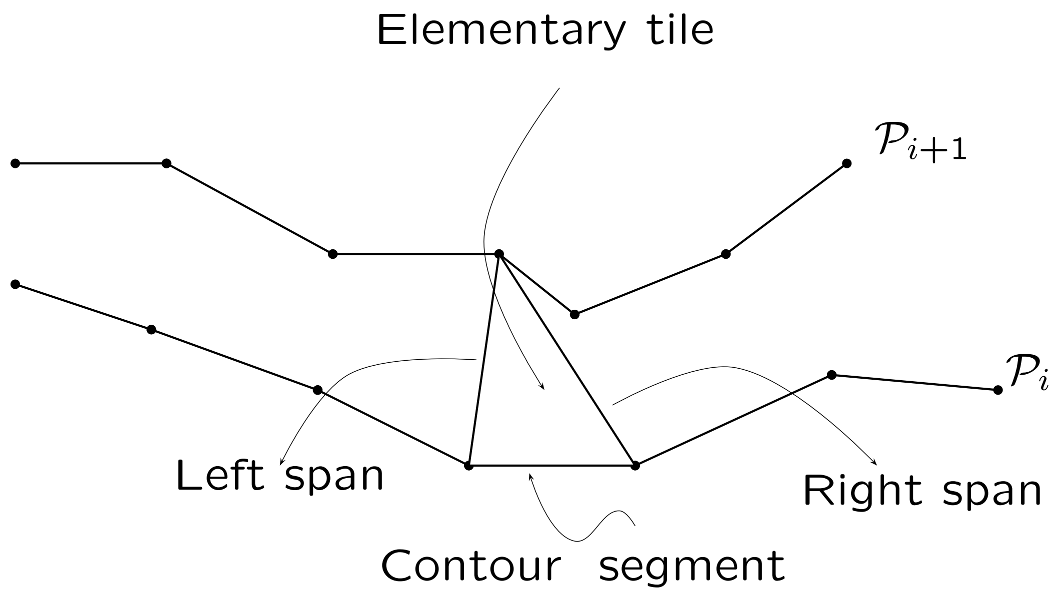
Suppose that there are arbitrarily scattered sample points $p_{ij} = (x_{ij}, y_{ij}, z_i)$ on a smooth surface Σ in $\mathbb{R}^3$, where $i = 1, 2 \cdots, M,\ j = 1, 2, \cdots, N_i$. Denote the set of all sample points by $\mathcal{P}$ and all sample points with level $z = z_i$ as $\mathcal{P}_i$, $i = 1, 2, \cdots, M$, where we assume $z_1 < z_2 < \cdots < z_M$. We call each set $\mathcal{P}_i$ as a cross section or a slice. Further, the set of all projections of p_{ij} onto xy plane is called a contour and denote it $\tilde{\mathcal{P}}_i$.

The problem to solve can then be summarized as the following questions. For given the set $\mathcal{P} = \bigcup_{i=1}^{M} \mathcal{P}_i$ of sample points,

- how can we reconstruct a piecewise interpolating curve from the set $\mathcal{P}_i$ of sample points with only geometric knowledge of the points $p_{ij} = (x_{ij}, y_{ij}, z_j)$?;

- how can we connect the p_{ij}'s and p_{i+1k}'s with straight lines in such a way as to form a triangular facet(or called an elementary tile) surface spanning two nested cross sections $\mathcal{P}_i$ and $\mathcal{P}_{i+1}$? This question is called a tiling problem.

More concisely, we set up the problem to solve following: A *contour segment* is a linear approximation of the curve connecting consecutive points in a single cross section. An *elementary tile(or triangular patches)* is a triangular face composed of a single contour segment and two spans connecting the end-points of a contour segment with a common point on the adjacent contour. The spans will be designated as “left” and “right” for obvious reasons.



Then the problem of reconstructing a 3D surface from a set of sample points on a series of parallel planar cross-sections is to find a set of elementary tiles which defines a surface satisfying two constraints:

(C1) Each contour segment will appear in exactly one elementary tile.

(C2) If a span appears as the left(right) span of some tile in the set, it will appear as the right (left) span exactly one other tile in the set.

A set of tiles which satisfy these two conditions is called an *acceptable surface*.

As described in the introduction, the main focus of this paper is to utilize the natural points ordering method reviewed in subsequent subsection and introduce a natural heuristic method for solving the described problem above. Therefore, we consider only the simple case that the set $\mathcal{P}$ of sample points satisfies the following restrictions:

- Sample points on each set $\mathcal{P}_i$ belong to a simple closed smooth space curve.

- Each cross section $\mathcal{P}_i$ consists of only one contour. That is, there are no branching problem.

- The interior of $\tilde{\mathcal{P}}_{i+1}$ is perfectly included in that of $\tilde{\mathcal{P}}_i$ for each i, where the interior of $\tilde{\mathcal{P}}_{i+1}$ means the interior set of the piecewise polygon constructed by connecting the ordered points of the set $\tilde{\mathcal{P}}_{i+1}$.

Natural Point ordering method

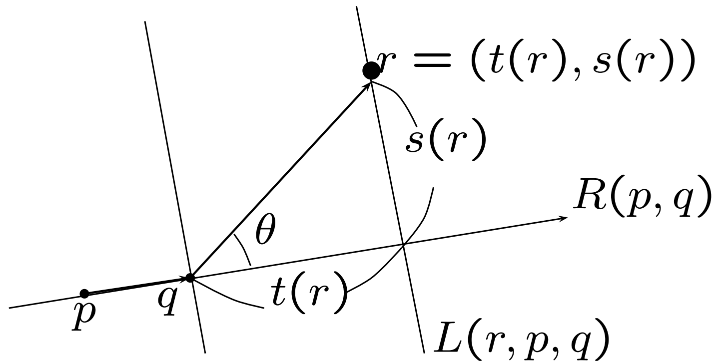
Natural distance. For given two points p and q in $\mathbb{R}^2$ which are already ordered in the direction $\overrightarrow{pq}$, the *natural distance* is an answer to the question: for each r which is on the opposite side of p, centered around the line perpendicular to $\overrightarrow{pq}$ containing q, how can we construct a distance between two points q and r which reflects the 'smoothness' of the piecewise linear arc $\widehat{pqr}$ and the 'closeness' between q and r simultaneously? In order to give a reasonable answer, we considered a small object moving from q in the direction of $\overrightarrow{pq}$ with constant velocity 1 and simultaneously diffusing randomly on the line which is perpendicular to $\overrightarrow{pq}$. A trace of such object corresponds to a graph of sample path of Brownian motion starting from q when we think of $\overrightarrow{pq}$ as a directional vector of time axis. The author then defined the *natural distance* $f(r)$ from q to r for directional vector $\overrightarrow{pq}$ as

$$f(r) = t(r) + K\frac{s(r)}{\sqrt{t(r)}}, \tag{1}$$

where

$$t(r) = |\overrightarrow{qr}|\cos\theta, \quad s(r) = |\overrightarrow{qr}|\sin\theta.$$

Here, θ $(-\frac{\pi}{2} < \theta < \frac{\pi}{2})$ is the signed angle between $\overrightarrow{pq}$ and $\overrightarrow{qr}$ defined by $\cos\theta = \frac{\overrightarrow{pq}\cdot\overrightarrow{qr}}{|\overrightarrow{pq}||\overrightarrow{qr}|}$



Then the first factor $t(r)$ was construed as the diffusion time of the small object from q to r for directional vector $\overrightarrow{pq}$ and call it the *time distance*, while the second factor $\dfrac{s(r)}{\sqrt{t(r)}}$ as the transition density measuring how smoothly moves the small object from q to r, and the probability that the small object moves from q to r with reaching time $t(r)$, and call it the *standardized probability distance*. For a detailed illustration, one refers to my paper. We call K the *subjective weight*, since it represents how much weight is given to the second factor. That is, the larger the subjective weight K is, the more sensitive than the natural distance it is to the magnitude of the change of the probability distance $\frac{s(\cdot)}{\sqrt{t(\cdot)}}$ than that of the time distance $t(\cdot)$.

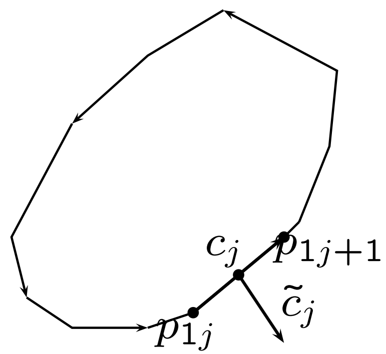
Natural points ordering method.

Using the natural distance described above, we developed a method to solve the curve reconstruction problem in $\mathbb{R}^2$ so called the *natural points ordering method*. It provides an algorithm to us for consecutive myopic choices of sample points in a natural way. In fact, it is designed to choose the points where the natural distance from the starting point(chosen one step before) is minimized.

A natural heuristic method

To describe the heuristic method we use, suppose we are given sample points distributed on two consecutive cross sections $\mathcal{P}_1$ and $\mathcal{P}_2$ as in section 1 and these set are well-ordered. Since both cross-sections are closed, we may think of the points p_{ij} as being extended periodically; i.e. $p_{1N_1+j} = p_{1j}$, $j = 1, 2 \cdots, N_1$ and $p_{2N_2+j} = p_{2j}$, $j = 1, 2, \cdots, N_2$. We assume that the cross-section $\mathcal{P}_2$ lies insides of $\mathcal{P}_1$ and the cross-sections $\mathcal{P}_i$, $i = 1, 2$, have counterclockwise orientation. For each index j, let c_j be the mid point of p_{1j} and p_{1j+1} and $\tilde{c}_j$ be the end points of the unit outward normal vector to the cross-section $\mathcal{P}_1$ with starting point c_j. Then these points c_j and $\tilde{c}_j$ are given by

$$c_j = (p_{1j} + p_{1j+1})/2, \quad \tilde{c}_j = (y_{1j+1} - y_{1j}, x_{1j} - x_{1j+1}, 0) + c_j, \quad j = 1, 2, \cdots, N_1.$$



For the simplicity of the notations, we let $\mathcal{T}_{ij}$ be the $(i+j)$th triangular facet. Then if the triangle $\mathcal{T}_{ij}$ has two vertices in $\mathcal{P}_1$, say $p_{1l},\ p_{1l+1}$, consecutively, and one vertex in $\mathcal{P}_2$, say p_{2m}, (it calls a triangle of Type I), then i and j mean $i = l-1$ and $j = m$, while if $\mathcal{T}_{ij}$ has two vertices in $\mathcal{P}_2$, say $p_{2l},\ p_{2l+1}$, consecutively, and one vertex in $\mathcal{P}_1$, say p_{1m}, (it calls a triangle of Type II), then $i = m-1$ and $j = l$. Further, let T_{kl} be the lth vertex of the triangle $\mathcal{T}_{ij}$ with $k = i+j$. As described in the following, we will directly construct the triangles of Type II from nested two triangles of Type I without any computation.

We are now ready to introduce our algorithm.

Initial step. To construct first triangular facet $\mathcal{T}_{01}$, start with $T_{11} = p_{11}$ and $T_{12} = p_{12}$. Then, by considering the direction vector $\overrightarrow{\tilde{c}_1 c_1}$, choose the third vertex T_{13} of $\mathcal{T}_{01}$ in the set $\mathcal{P}_2$ satisfying

$$T_{13} = \underset{r \in \mathcal{P}_2}{\operatorname{argmin}} f(r),$$

where $f(\cdot)$ is the natural distance defined in (1).

Once the vertex T_{13} is chosen, then we reorder the ordered set $\mathcal{P}_2$ starting with T_{13} and we let $T_{13} = p_{21} = p_{21^{1}}$.

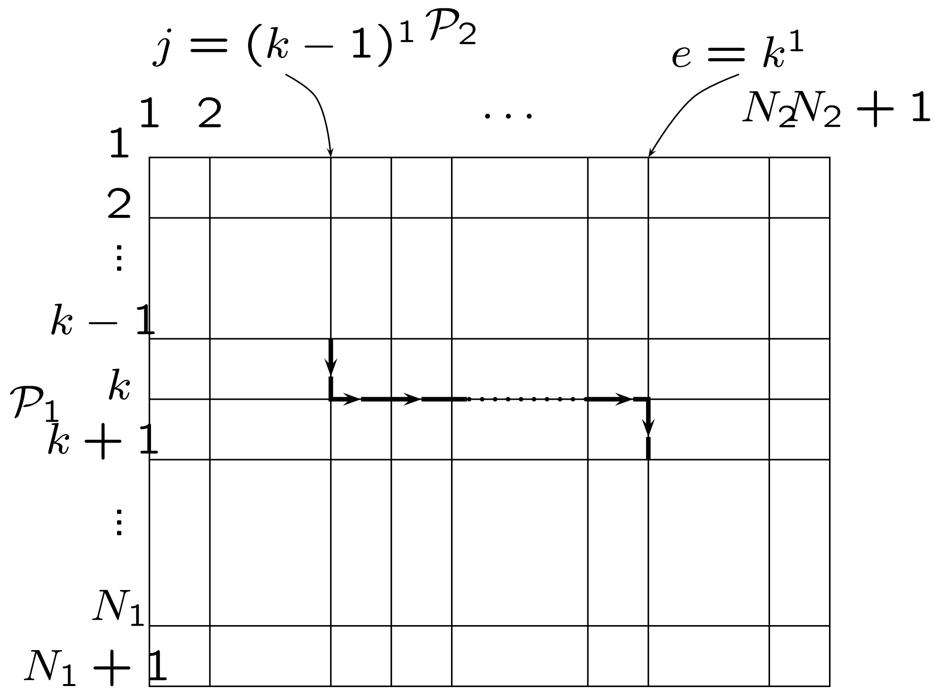
Middle steps. Assume that p_{2j^1}, $2 \leq j \leq N_1$ have already been chosen with the following law, where $p_{2j^1} \in \mathcal{P}_2$ and $1 = 1^1 \leq 2^1 \leq \cdots \leq j^1$, and let $C_{j+1} = \mathcal{P}_2 \setminus \{p_{21^1}, p_{22^1}, \cdots, p_{2(j-1)^1}\}$. We choose the point $p_{2(j+1)^1}$ in the set C_{j+1} satisfying

$$p_{2(j+1)^1} = \underset{r \in C_{j+1}}{\text{argmin}} f(r),$$

where we assume that the initial direction vector is $\overrightarrow{\tilde{c}_{j+1} c_{j+1}}$.

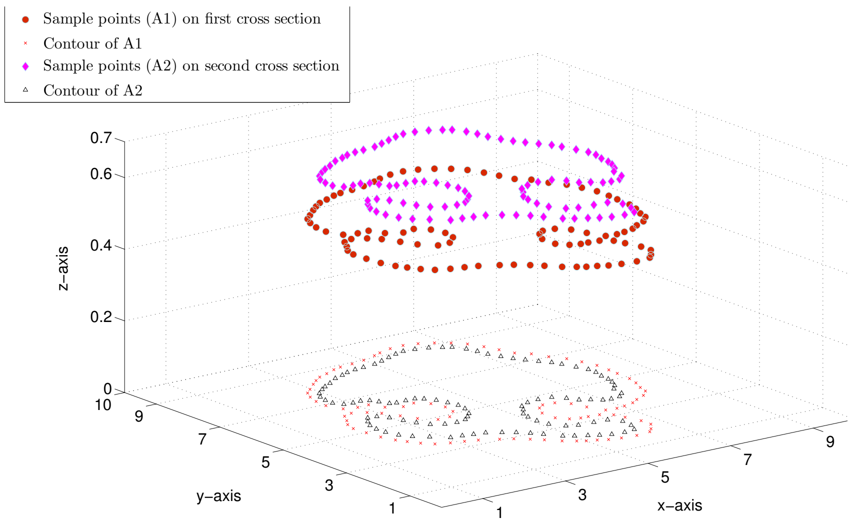
Final step. Assume that $\mathcal{T}_{01}, \mathcal{T}_{11}, \cdots, \mathcal{T}_{1m_1}, \cdots, \mathcal{T}_{ln}, \cdots, \mathcal{T}_{ln_l}, \cdots, \mathcal{T}_{k-2j}$, $2 \leq k \leq N_1$, $j \geq 1$, $j = (k-1)^1$, have already been constructed, where $\mathcal{T}_{k-2j}$ is a triangle of Type I. Then three vertices of $\mathcal{T}_{k-2j}$ are p_{1k-1}, p_{1k}, p_{2j}. Now let $e = k^1$. If $e = j$, we construct the next triangle as $\mathcal{T}_{k-1e}$ with vertices $T_{k-1+e1} = p_{1k}$, $T_{k-1+e2} = p_{1k+1}$ and $T_{k-1+e3} = p_{2e}$. When $e > j$, first construct triangles of Type II, $\mathcal{T}_{k-1j}, \mathcal{T}_{k-1j+1}, \cdots, \mathcal{T}_{k-1e-1}$ and then construct a triangle of Type I, $\mathcal{T}_{k-1e}$ (see Figure **??**).

Finally, we consider the final triangle facet $\mathcal{T}_{N_1-1e}$. If $e = N_2 + 1$, then stop the process, while if not, we add further triangle facets of Type II, $\mathcal{T}_{N_1 e}, \mathcal{T}_{N_1, e+1}, \cdots, \mathcal{T}_{N_1 N_2}$.

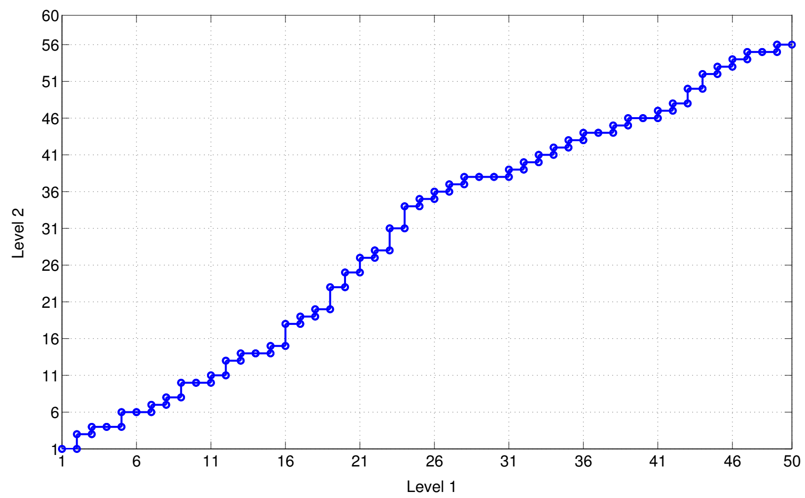


$j = (k-1)^1$
$\mathcal{P}_2$
$e = k^1$
1
2
...
N_2
$N_2 + 1$
1
2
$k-1$
k
$\mathcal{P}_1$
$k+1$
N_1
$N_1 + 1$

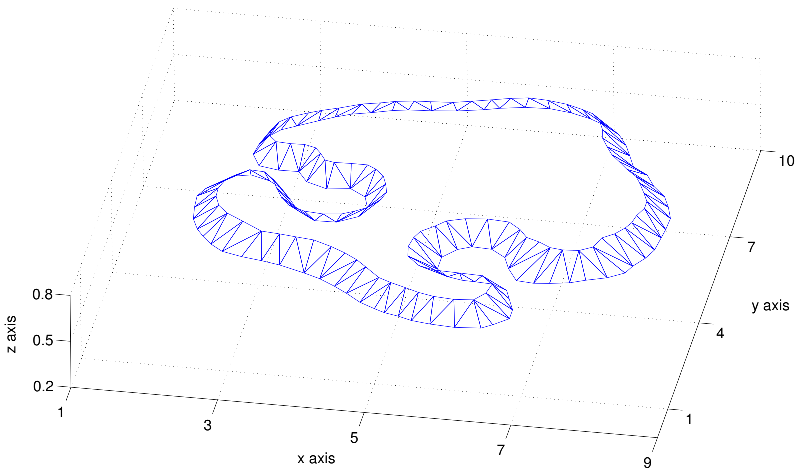
Simulation Results



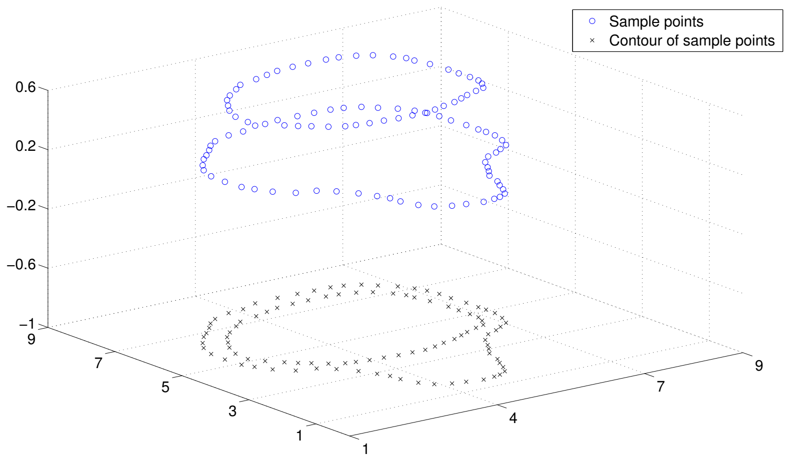
(a)



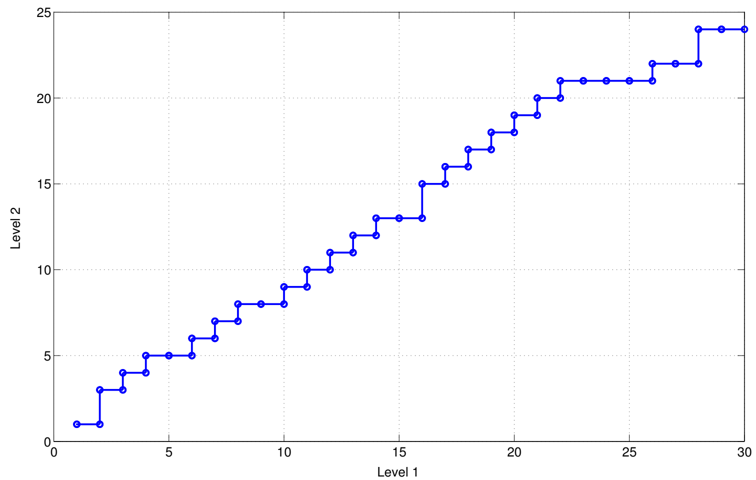
(b)



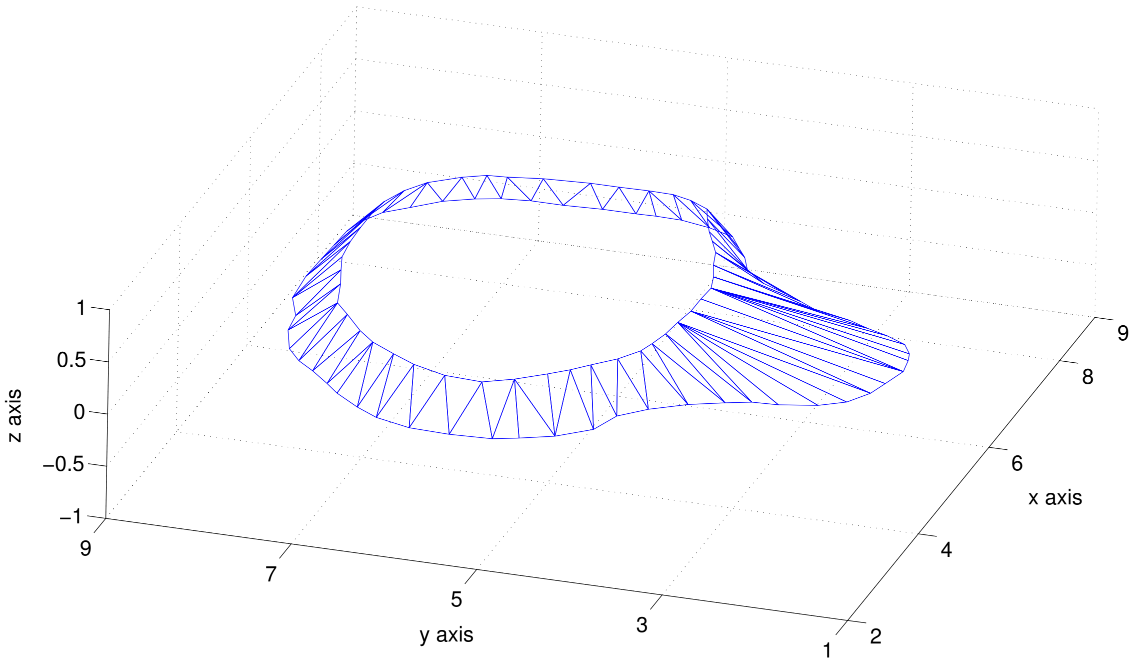
(c)

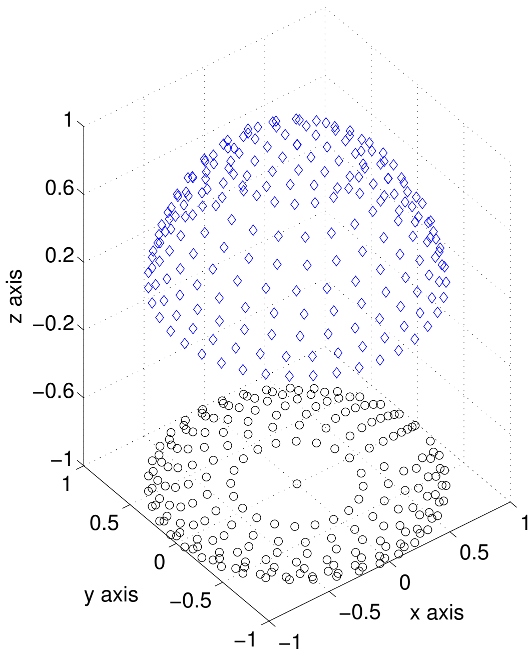

Sample points
Contour of sample points
0.6
0.2
−0.2
−0.6
−1
9
7
5
3
1
1
4
7
9

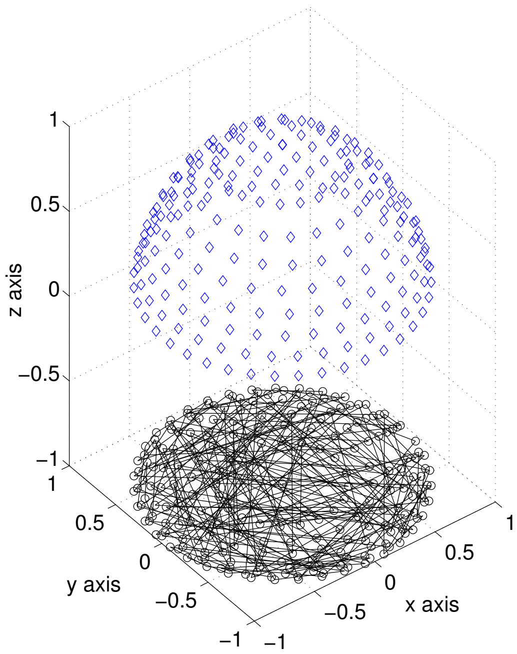
(a)

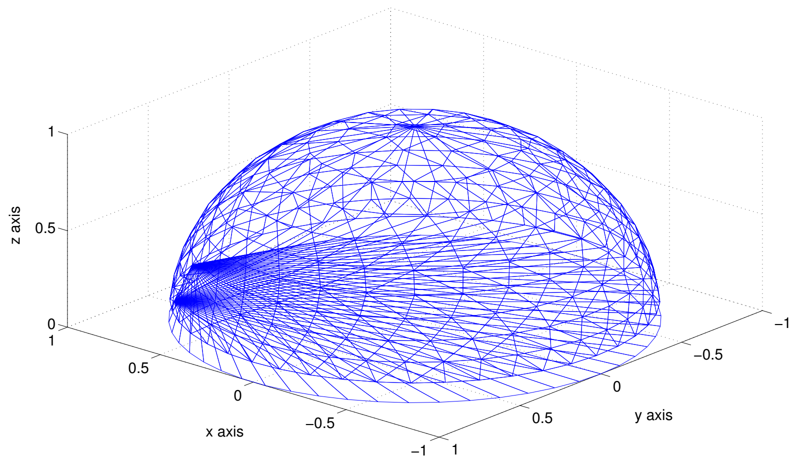

Level 2
Level 1
0
5
10
15
20
25
30

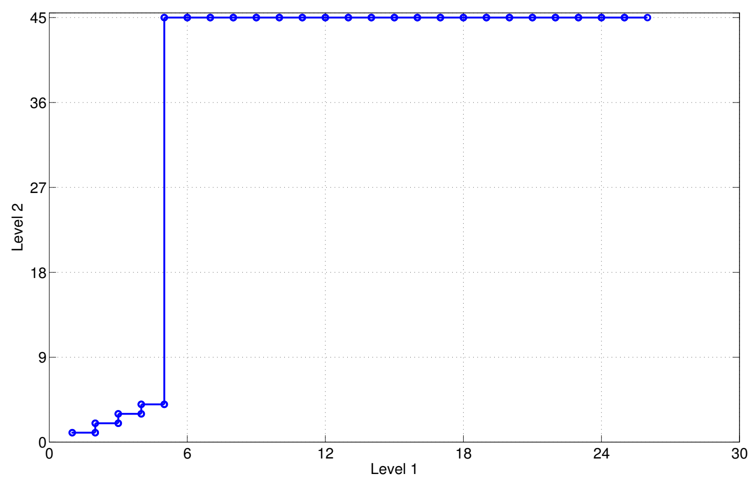
(b)

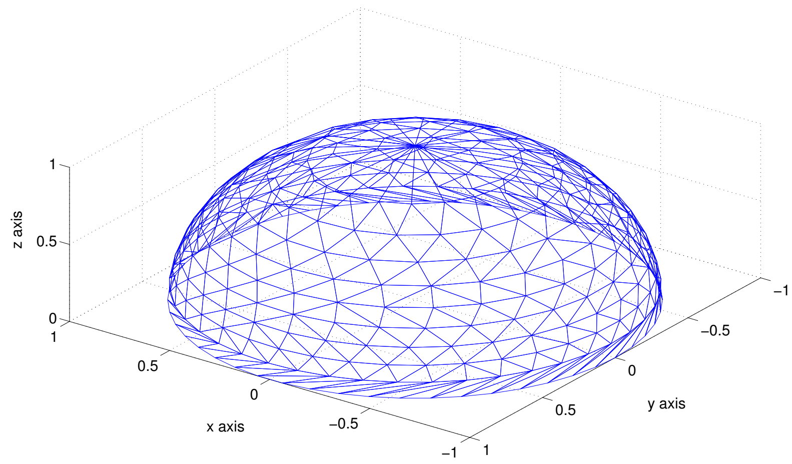

1
0.5
0
−0.5
−1
z axis
9
7
5
3
1
y axis
2
4
6
8
9
x axis

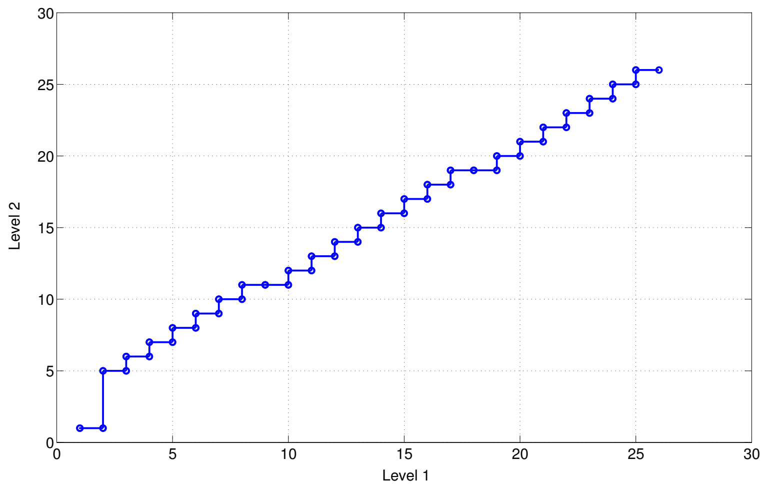

1
0.6
0.2
−0.2
−0.6
−1
z axis
1
0.5
0
−0.5
−1
y axis
−1
−0.5
0
0.5
1
x axis


1
0.5
0
−0.5
−1
z axis
1
0.5
0
−0.5
−1
y axis
−1
−0.5
0
0.5
1
x axis


z axis
x axis
y axis
0
0.5
1
0.5
0
−0.5
−1
1
0.5
0
−0.5
−1


Level 2
Level 1
0
9
18
27
36
45
0
6
12
18
24
30


z axis
x axis
y axis
1
0.5
0
-0.5
-1


Level 2
Level 1
0
5
10
15
20
25
30