{file watching.tex August 4, 1998 – for *Machine Graphics & Vision*, Aug. 1998}

Watching steps of a grand tour implementation

Anna Bartkowiak and Adam Szustalewicz
Institute of Computer Science, University of Wrocław
Przesmyckiego 20, Wrocław, 51–151 Poland
e-mail: aba, asz@ii.uni.wroc.pl

Abstract. Bartkowiak and Szustalewicz, 1997 (B&Sz), have proposed an implementation of the grand tour algorithm designed specially to detect outliers in multivariate data. The algorithm works by performing a sequence of rotations and projections onto a specific 2D-plane (Π_{proj}). This is equivalent to perform a series of rotations yielding transformed coordinates of the data: $\mathbf{X}^{(tr)} = \mathbf{XA}$, with $\mathbf{A}$ being the rotation matrix, while $\mathbf{X}$ and $\mathbf{X}^{(tr)}$ denote the data matrices before and after the rotation appropriately. A superimposed concentration ellipse indicates the outstanding points.

We complement the paper of B&Sz by considering some details and variants of the implementation of the grand tour algorithm. In particular we watch the trajectories of the projected points. Our concern is the denseness of the watched projections. We look at the trajectories of the projected points visible in the Π_{proj} plane and the frequencies of their appearing in various sectors of that plane.

In the *Appendix* we present the derivation of the formula for the rotation matrix $\mathbf{A}$ employed in each step of the algorithm for obtaining the transformed data coordinates.

Key Words. Dynamic graphics, Grand tour, Rotation, Projection, Uniform distribution.

1 Introduction

The paper considers some aspects of the grand tour algorithm implementation by Bartkowiak and Szustalewicz [5], called hereafter B&Sz. The algorithm yields dynamically changing projections, which are observed in a specific plane Π_{proj}. We complement the paper by B&Sz by considering some details and variants of the implementation of the grand tour algorithm. We consider it to be relevant, because the real usefulness of the algorithm depends on the details of its implementation. Therefore we discuss step-by-step the details of our implementation of the grand tour algorithm. In particular we verify, (*i*) whether the observed projections are becoming dense in the space of all possible projections of the kind used, and (*ii*) how fast the representativeness is attained.

In *Section 2* the grand tour algorithm is briefly reconsidered and the main principles of the work of the algorithm are recalled.

Section 3 shows an application of the grand tour algorithm to a real data set, namely the farm data [4], containing 5 economical indices for 31 farms growing vegetables. The data set can be imagined as a set of $n = 31$ points located in a $p = 5$ dimensional feature space R^p.

Some considerations on additional details of performing the grand tour are shortly presented in *Section 4*.

In *Section 5* the trajectories of points and concentration ellipses – as observed in the Π_{proj} plane – are receiving more attention. We consider here a full cycle rotation performed in such a way, that – for randomly chosen direction $\overrightarrow{UV}$ – we carry out a rotation of the data cloud by 360^o. We retain in the projection plane intermediary steps obtained after sequential rotations by a small angle α (we consider $\alpha = 9^o$ and $\alpha = 15^o$). The small angles add up to the full rotation of the data cloud (i.e. rotation by $\alpha = 360^o$). We present views obtained for some selected data vectors of the farm data when carry out such full rotations.

In *Section 6* the frequency of appearing of the projected points – in the centered sectors of the plane Π_{proj} – is considered. We expect here that the frequency of points appearing in subsequent sectors will be distributed uniformly over the sectors. For the considered farm data ($p = 5$), in principle, about 120 projections were sufficient to cover uniformly all the sectors of the projection plane. However, with a larger number of projections, we stated some deviations from the uniformity of the distribution.

Some conclusions are given in *Section 7*.

The algorithm is implemented in XLisp-Stat [2]. Also, all graphs were obtained using that system.

2 Principles of the grand tour algorithm

The grand tour is a way of viewing multivariate data points located in a truly multivariate space with dimension greater than 3. The concept and expression *grand tour* was introduced by Asimov [1]. The method was implemented in the C language as the program *XGobi* by Swayne, Cook and Buja [3] and in the XLispStat system by L. Tierney [2]. An implementation with the special goal of detecting atypic data vectors was proposed by Bartkowiak and Szustalewicz (B&Sz) in [5].

The approach used by B&Sz is illustrated in Figure 1. We see there a sphere representing the p-variate ($p \geq 3$) hypersphere. The center of the sphere is also the center of the coordinate system. The hypersphere encloses a data cloud containing n points (for clarity of the graphical presentation the data points are not shown in that figure).

Let now suppose that there is an observer located somewhere outside the sphere. At any moment the observer may have a view on all the data points. His viewing is equivalent to watch the projections of the data points onto some two–dimensional projection plane, which (i) is perpendicular to the direction of the observers look, and (ii) passes through the center O.

E.g., let us consider the situation presented in Figure 1. Suppose that the observer is located outside the sphere somewhere on the (prolongation) of the x_3 axis. Then he will see all the points of the data cloud projected onto a plane parallel to the $<x_1, x_2>$ plane. When the observer diverges more and more to infinity along the x_3 axis, then the seen by him point projections will approach the x_1, x_2 coordinates of the viewed points.

To obtain different views, the observer has either (a) to move round the sphere,

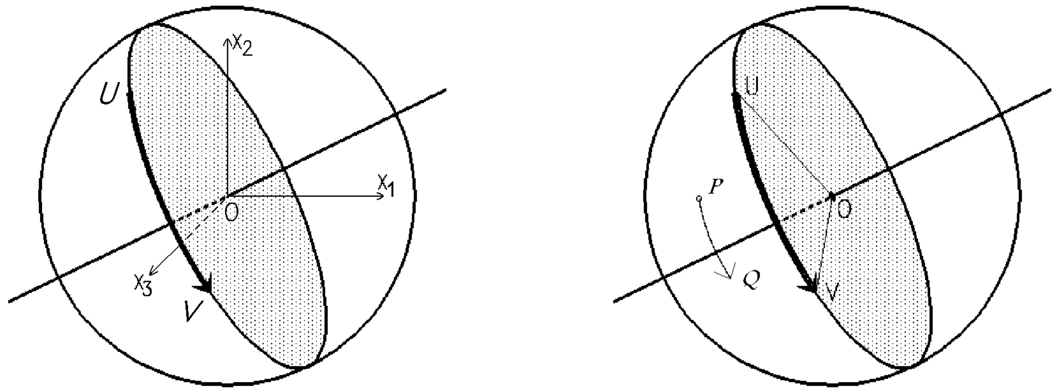

Figure 1. Hypersphere centered at O with coordinate system also centered at O.

Left: Grabbing the hypersphere at some randomly chosen point U and moving the sphere along a geodetic path up to the point V designates a 2D-plane containing also the center O.

Right: Rotating an arbitrary point P around the normal to the 2D-plane (established by the points U, V, O) – to attain its new position Q.

or (b) to rotate the sphere. In both cases the observer sees projections of the data points onto some two–dimensional planes.

In the following we will consider rotations of the sphere. By a rotation of the sphere we will mean grabbing at a point U on the hypersphere and pushing the sphere in a randomly selected direction given by another point V on the hypersphere, however keeping the center of the sphere unmoved. In such situation, with each rotation is connected a plane passing through the center of the sphere and a geodetic path on the surface of the hypersphere – which is illustrated in Figure 1.

The rotations, carried out in sequence, produce diverse views of the data cloud.

The main postulates of such data viewing are:

- The views should become rapidly dense in the space of all views possible in the described circumstances. By this we mean that the rotations should be done in directions constituting a sort of representation of all possible directions in the considered R^p space. In other words, our wish is to obtain views when looking from possibly almost *all* 'sides' of the data cloud.

- The changes (passages) from one view to the another should be performed smoothly, thus a kind of plot interpolation is needed.

These postulates open field for various realizations of the grand tour.

The approach chosen by B&Sz may be summarized as follows. Firstly a plane of rotation, Π_{rot}, is established. This plane is identified uniquely by two points, U and V, located on the sphere, and the center O, through which it passes (see Figure 1). In the following it will be called *the reference plane for the rotation* and denoted as $<UOV>$ or Π_{rot}.

Now suppose, we rotate the sphere together with the data points in such way that we keep the complement to the plane <UOV> constant (for $p = 3$ this is a rotation around the *normal* to the plane <UOV> passing through O). We assume that the rotation does not affect the coordinate system and keeps it unmoved. Obviously the coordinates of the points – referred to the unmoved coordinate system – are now different than those before the rotation.

It is known, that the 'new' coordinates (after the rotation) may be obtained as a transformation of the 'old' ones (i.e. given before the rotation). Let $\mathbf{X}$ and $\mathbf{X}^{(tr)}$ of size $n \times p$ denote the data matrices (containing coordinates of the data points), before and after the transformation appropriately, with n denoting the number of data points (individuals), and p equal to the number of features recorded for each individual. Then $\mathbf{X}^{(tr)}$ is obtained from $\mathbf{X}$ using the formula:

$$\mathbf{X}^{(tr)} = \mathbf{X}\mathbf{A}, \tag{1}$$

where $\mathbf{A}$ of size $p \times p$ is the *transformation* matrix, called also *rotation* matrix. The matrix $\mathbf{A}$ is fully established by the points U, V, O and by the rotation angle, by which we rotate the data cloud. The rotation angle might be taken as the angle $\measuredangle\, UOV$, or any other angle α, $0^o \leq \alpha < 360^o$. Below we give a formula for constructing the rotation matrix $\mathbf{A}$ for given data points U, V and given angle α.

Let $\mathbf{u} = (u_1, \ldots, u_p)$ and $\mathbf{v} = (v_1, \ldots, v_p)$ denote the vectors containing the normalized coordinates of the points U, V (*normalized* means, that the respective vectors $\mathbf{u}$, $\mathbf{v}$ are of unit length).

We firstly construct a vector $\mathbf{t} = (t_1, \ldots, t_p)$ of unit length, orthogonal to the vector $\mathbf{u}$ and lying in the plane <UOV> (this is done using one step of the Gram-Schmidt orthogonalization). The elements a_{ij} of the rotation matrix $\mathbf{A}$ are given by the formula:

$$a_{ij} = \delta_{ij} + u_i * [u_j * (\cos(\alpha) - 1) + t_j * \sin(\alpha)] + t_i * [-u_j * \sin(\alpha) + t_j * (\cos(\alpha) - 1)]. \tag{2}$$

The derivation of the presented formula is shown in the *Appendix*.

Let us emphasize, that the rotation matrix $\mathbf{A}$ is a function of α, the given rotation angle. Thus

$$\mathbf{A} = \mathbf{A}(\alpha) = (a_{ij}(\alpha)), \qquad i, j = 1, \ldots, p.$$

Dealing with the data matrix $\mathbf{X}$ of size $n \times p$ we may apply the transformation $\mathbf{A}$ separately to each individual data point (row data vector) or to the entire data matrix, thus obtaining:

$$\mathbf{X}^{(tr)}_{n\times p} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \ldots \\ \mathbf{x}_n \end{bmatrix}_{n\times p} \mathbf{A}_{p\times p}. \tag{3}$$

The matrix $\mathbf{X}^{(tr)}_{n\times p}$ contains (in rows) the coordinates of the rotated data points $\mathbf{x}_1^{(tr)}, \mathbf{x}_2^{(tr)}, \ldots, \mathbf{x}_n^{(tr)}$ expressed with regard to the original (unrotated) coordinate system. Obviously

$$\mathbf{x}_i^{(tr)} = \mathbf{x}_i\mathbf{A}.$$

After rotating the data cloud we might project the rotated points onto any particular plane. We choose the particular projection given by the matrix

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}.$$

Taking such projection we do not need any additional calculations to obtain the coordinates of the projected points: these are simply the first two columns of $\mathbf{X}^{(tr)}$. Thus:

$$\Pi_{proj} = <x_1, x_2> .$$

We obtain then

$$\tilde{\mathbf{X}}^{(tr)}_{n\times 2} = \mathbf{X}^{(tr)}_{n\times p}\mathbf{P}_{p\times p},$$

with $\tilde{\mathbf{X}}^{(tr)}_{n\times 2}$ containing the first two columns of the transformed data matrix dealt with.

Constructing subsequently for steps $k = 1, \dots, K$ the matrices $\mathbf{A}^{(1)}, \mathbf{A}^{(2)}, \dots, \mathbf{A}^{(K)}$ designating the rotations and applying them to the (subsequently being transformed) data matrices, we obtain – starting with $\mathbf{X}^{(0)}$

$$\mathbf{X}^{(K)} = (\dots(((\mathbf{X}^{(0)})\mathbf{A}^{(1)})\mathbf{A}^{(2)}) \dots)\mathbf{A}^{(K)}. \tag{4}$$

In each step we take from each (transformed) data matrix the first two columns and use them for constructing a dynamic scatterplot, which means that after each rotation we clear the previous image in the $<x_1, x_2>$ plane and draw a new scatterplot on the basis of the first two columns of the matrix $\mathbf{X}^{(k)}$ obtained from the utmost recently carried out transformation.

For practical reasons it is advisable to consider *standardized* data matrices. This means, that from each data value x_{ij} the respective mean $\overline{x}_{.j}$ is subtracted, and the result is divided by the standard deviation s_j; the index j denotes here the jth variable, and the index i the *no.* of the data point: $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})$, for $i = 1, \dots, n$.

The projection plane considered when using the standardized data will be denoted in the following as $<\tilde{x}_1, \tilde{x}_2>$.

3 The grand tour applied to a real data set: the farm data

Now we will illustrate the grand tour method – as presented in previous section – by applying it to a real data set, namely the *farm data*, which were analyzed a.o. in [4]. The data set contains some economic indices and vegetable production of 31 Polish farms.

We will take for our illustration the 5 economic indices. Thus we will consider a data matrix $\mathbf{X}$ of size 31×5. Each farm will be represented by a data vector $\mathbf{x}_i = (x_{i1}, \dots, x_{i5})$, $i = 1, \dots, 31$. In the following $\mathbf{X}$ will denote the standardized data matrix. We will also assume, that the horizontal and vertical axes of the projection plane will contain coordinates given by the first and the second columns of the transformed data.

After choosing two random vectors $\overrightarrow{OU}$ and $\overrightarrow{OV}$, constructing from them the rotation matrix **A**, and making the projection onto the plane spanned by the horizontal and vertical axes of the screen, we obtained the view shown in Figure 2, upper row (figure drawn by XLispStat with numeration starting from 0!).

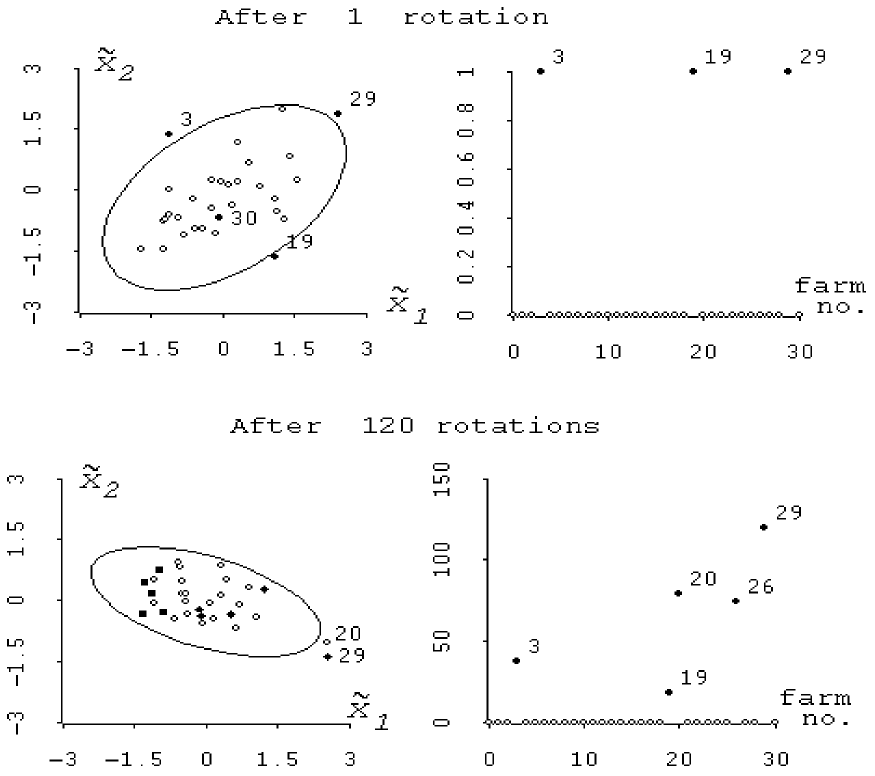

Figure 2. Exemplary projections obtained after applying one (top row) and 120 (bottom row) rotations to the farm data. The $\tilde{x}_1$ and $\tilde{x}_2$ axes denote appropriately the horizontal and vertical axes of the projection plane.
Left: Scatterplots containing projections of points with 95% concentration ellipse superimposed.
Right: A linked count plot recording, how many times each data point (marked in the farm no. axis) has fallen beyond the borders of the concentration ellipse. For the situation shown in the upper plot, there was only one projection, thus the count values are '1' for the points # 3, 19, 29, and '0' for the remaining points. Notice, that **point numeration (#) starts from 0!**

To detect outliers we superimpose on the scatterplot (containing the point projections) a concentration ellipse. This may be done in several ways.

Generally, the shape and orientation of the ellipse is designated by the covariance $cov(\tilde{x}_1, \tilde{x}_2)$ and the variances $var(\tilde{x}_1)$ and $var(\tilde{x}_2)$ of the coordinates of the point projections. The size of the ellipse is designated by a parameter β $(0 \le \beta \le 1)$.

The concentration ellipse is supposed to cover in average $\beta \times 100\%$ of points coming from a bivariate normal distribution with covariance $cov(\tilde{x}_1, \tilde{x}_2)$ and variances $var(\tilde{x}_1)$ and $var(\tilde{x}_2)$.

The ellipses visible in the left plots of Figure 2 were simply evaluated from the coordinates of the point projections visible in the respective plots, taking $\beta = 0.95$.

Looking at the ellipse visible in the upper left plot of Figure 2 one can see, that 3 points: # 3, 19 and 29 (in the numeration starting from 0!) are located outside the concentration ellipse. This makes them suspected as 'being outliers' (for some concepts, what is meant by being an outlier, see. e.g. [5] or [6]).

After obtaining the view in the top row of Figure 2 we have continued the grand tour by performing additional 119 rotations. Each rotation was constructed from anew generated pair $\overrightarrow{OU}, \overrightarrow{OV}$ and a fixed angle 20^o.

The views obtained after performing the 120 rotations are shown in Figure 2, bottom row. In the left plot we see the outstanding position of points # 20 and 29. The right count plot exhibits a kind of summary of our search for outliers: Points # 3, 19, 20, 26 and 29 were quite often notified as located outside the concentration ellipse.

4 Some additional details on performing the rotations

Suppose, that we have generated two vectors $\overrightarrow{OU}$ and $\overrightarrow{OV}$ establishing the reference plane $\Pi_{rot} = <UOV>$ for the rotation. Then we have still to fix the angle α of the rotation (for a given point P the angle α designates the point Q attained from P after rotation – see Figure 1, right plot).

We could proceed here in the following ways:

- take as the rotation angle α simply the angle UOV (i.e. $\alpha = \measuredangle UOV$),
- take as the rotation angle α a small given angle α_1, e.g. $\alpha_1 = 9^o$ or $\alpha_1 = 15^o$.

Both cases allow us to construct the rotation matrix $\mathbf{A}$ according to formula (2). Remind, that $\mathbf{A}$ is used for transforming the data matrix $\mathbf{X}$ according to formula (3).

However, why should we use $\mathbf{A}$ only once? The established transformation $\mathbf{A}$ might be applied to the data matrix $\mathbf{X}$ sequentially several times, say K times, as it was assumed and shown in formula (4). If so, then each time the data points would be rotated with regard to the same reference plane $<UOV>$. Choosing small angle α_1 and rotating several times with regard to the once established reference plane $<UOV>$ will give smooth transition from one position of the points to the other.

In the following two sections we will observe in more details the trajectories of the points when composing the summary rotation from small angles α_1 (the trajectories of the watched points will appear then as nearly smooth lines).

Firstly, in Section 5 we will watch the trajectories, when making – for a given reference plane – a full cycle of rotations, i.e. when rotating the points by small angles adding up to 360^o.

Next, in Section 6 we will watch the trajectories, when making – for a given reference plane – several rotations by a small angle α_1, adding to a summary angle γ, with $\gamma < 360^o$.

To make the images of individual points clearly visible and not overlaid, we have chosen ten exemplary points for which we pursue individually their trajectories seen during 5 subsequent cycles of full rotations.

5 Trajectories of points observed in the projection plane when making a full cycle of rotation

In this section we will consider in more detail a full cycle of rotation, by which we mean, that having chosen the reference plane for the rotation (for example such as given by the points U, V, O in Figure 1) we rotate the hypersphere – together with the data cloud – by 360^o.

To achieve such full cycle we may compose it from smaller angles, adding to 360^0. In the following we will pay our attention to the case when the full cycle is achieved by adding 40 times successive rotations by the same angle $\alpha = 9^o$. After each successive rotation we will observe and record the image visible in the projection plane Π_{proj}.

We will firstly observe the trajectories of *two representative data points* from the farm data. Next we will observe the trajectories of a subset containing *ten data points* – chosen also from the farm data.

5.1 Pursuing individual trajectories for 2 selected data points

For watching individual trajectories we have chosen *two* points from the farm data. These are points # 29 and # 30 (in the numeration starting from 0), both visible in Figure 2. The point # 29 seems to be an outlier, while point # 30 seems to occupy a central position in the analyzed data cloud.

In Figure 3 we show – for both points – the trajectories of the projections notified during 5 subsequent full rotations by 360^o. Each full rotation was composed from 40 small rotations by 9^o. The rotations during each cycle were carried out using for both points the same matrices $\mathbf{A}^{(1)}, \ldots, \mathbf{A}^{(5)}$. This means that each of the matrices $\mathbf{A}^{(i)}$ $(i = 1, \ldots, 5)$ was obtained by formula (2) from anew generated points $U^{(i)}, V^{(i)}$ and the same assumed angle $\alpha = 9^o$.

For each point the trajectory of the given point projection (#29 or #30) – as stepped during one full cycle – is shown in a separate subplot of Figure 3. Thus for 5 cycles we got 5 subplots for each point. The 6th subplot is a superposition of the former 5 subplots.

Looking at the individual trajectories (displayed in Figure 3) one may see, that – with increasing number of steps – the point projections, visible in the Π_{proj} plane, move by small increments. In every case the position of the point projection after next step depends heavily on its position after the previous step. Generally, making rotation by an angle $\measuredangle(k * \alpha_1)$ in the R^p space – according to some reference plane designating the direction of the rotation, (as e.g. that shown in Figure 1) – and taking both $k > 0$ and α_1 small – the point projection visible in Π_{proj} will not move far from its position for $k = 0$. In such case we might loose some interesting views, which could be picked up when making a rotation with a larger angle (larger k).

On the other hand, making full rotation – with the aim to gather all the interesting views, if any – might be non economical (remember, that in any case we should achieve the rotation by a larger angle carrying out small incremental steps to preserve the smoothness of the movements in the Π_{proj} plane).

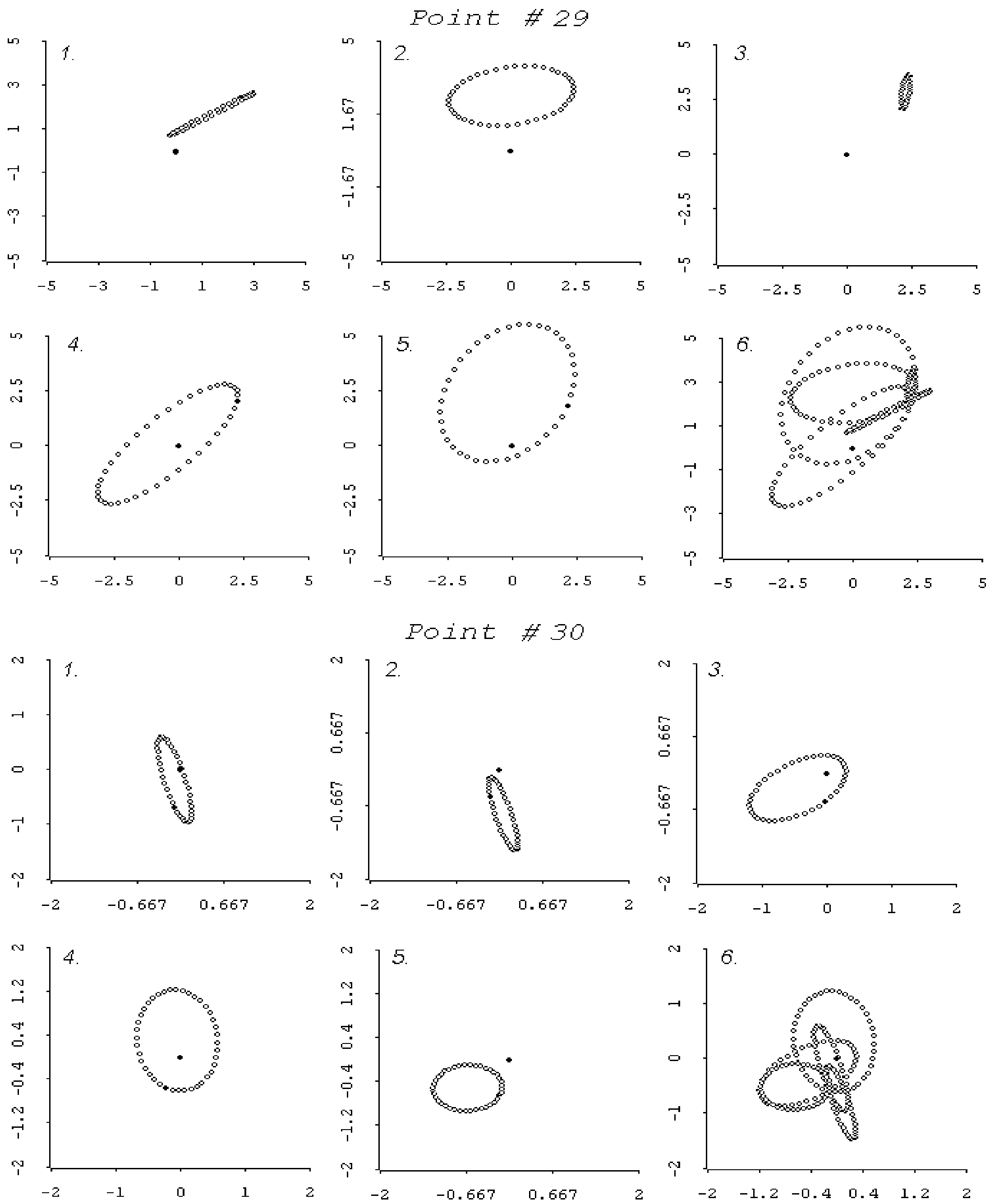

Figure 3. Individual trajectories of points # 29 and # 30 – observed in the $<\tilde{x}_1, \tilde{x}_2>$ *plane – when rotating 40 times the data cloud by the angle* $\alpha = 9°$, *thus making a full cycle of rotation. Presented are trajectories observed during five subsequent full cycles. First 5 subplots for each point show point trajectory during one cycle. The 6th subplot shows superposition of the five preceding subplots. It might be interesting to notice the difference in the range of the coordinates for point # 29 and # 30.*

Therefore we have to decide, how to choose the angle of rotation, to obtain a dense set of projections covering possibly many diversified directions of projections.

We will try to say more on this topic in Section 6.

5.2 Global transitions from one stage to another – observed for a selected subset of the farm data

For a more detailed watching of the trajectories of the point projections and of the ellipses – when performing subsequent rotations – we have chosen a subset comprising 10 points of the analyzed data set, namely the points (#0–#4, #26–#30). The chosen subset is shown in Figure 4.

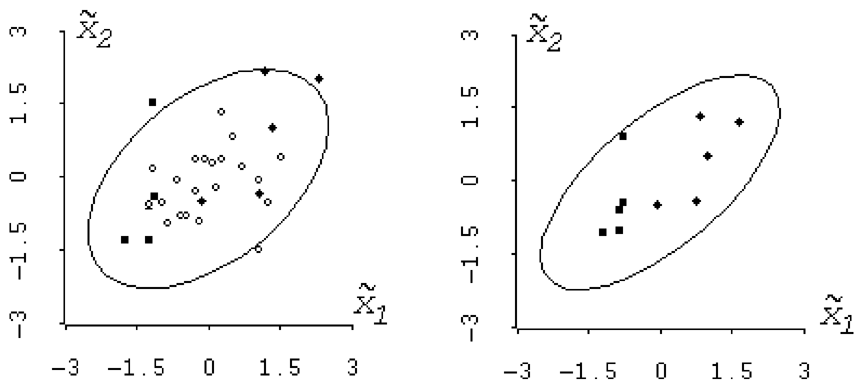

Figure 4. Choosing a subset from the farm data.
Left: The entire data cloud containing $n = 31$ points. Ten points are highlighted. Points # 0–4 are denoted by filled squares, # 5–25 by empty circles, # 26–30 by filled diamonds.
Right: Only the 10 points selected. The ellipse calculated from the 10 chosen points has a different shape, because the variances and the covariance in the subset are different from those evaluated from the entire data set. Also, both data sets were standardized using different means and different standard deviations.

Consider now the situation when we have already fixed the direction of the rotation, i.e. the reference plane for the rotation. We want to have a smooth transition of points depicted in Π_{proj} from one position (before the successive rotation) to the next (after the rotation performed). To achieve this, we should make the rotation by small angles α_1.

Suppose that we rotate the data cloud by the angle $\alpha_1 = 15^o$ in the direction established by $\overrightarrow{UV}$. Then, repeating the rotation (in the same direction) 24 times, we will obtain a full cycle rotation by 360^o – with the data cloud and the point projections returned to their original position. This may be seen in Figure 5.

The figure contains 6 subplots (groups) called in the following *fivnums*.

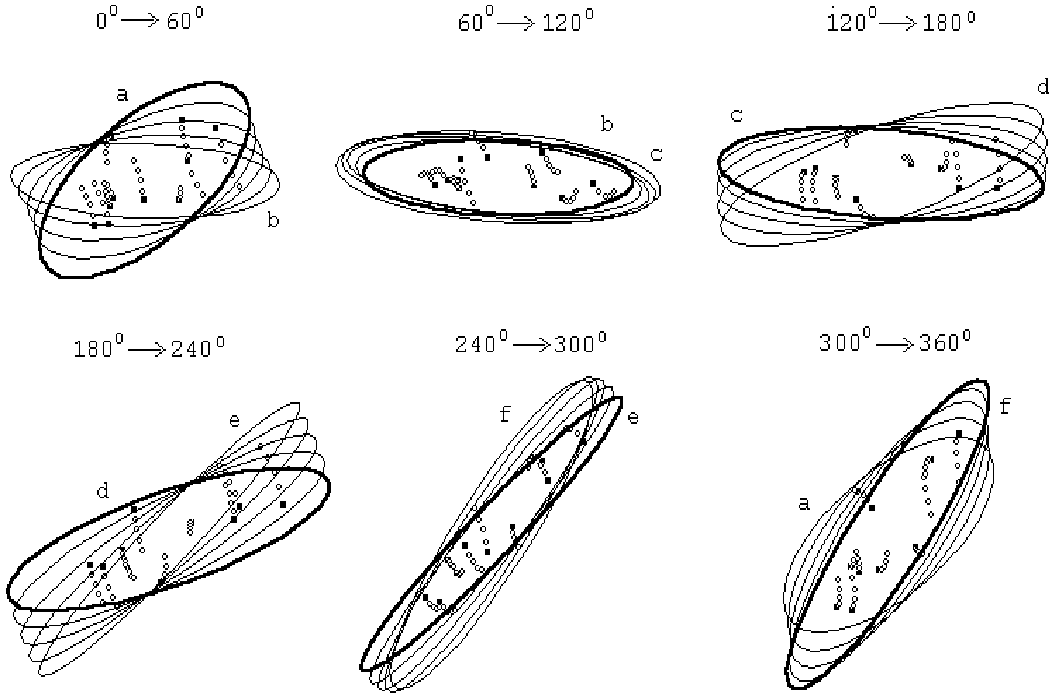

Figure 5. Trajectories of points and ellipses of concentration, when rotating the data cloud by the angle $\alpha_1 = 15°$; captured in groups – fivnums. Each fivnum shows point projections and corresponding to them concentration ellipses at the start and after 4 subsequent rotations by 15^o. Ellipse first in the fivnum is enhanced. Points being in the starting position in each fivnum are highlighted and denoted by filled squares. The axes $\tilde{x}_1, \tilde{x}_2$ for each fivnum, the same as in Figure 4, were omitted for clarity of the presentation.

Each fivnum is composed from 5 ellipses and 5 sets of point projections.

In the first row of Figure 5 we have from left to right the views after rotating the data points by 60^o, 120^o, 180^o. Then, in the bottom row: positions attained after rotation by 240^o, 300^o, and 360^o respectively. The last projection has completed the full cycle – thus its image coincides with the starting ellipse and the starting (filled) points presented in the first fivnum.

Each fivnum exhibits a starting view (of point projections and their concentration ellipse) and the respective views obtained from 4 subsequent rotations by the angle $\alpha_1 = 15^o$ – without clearing the preceding images in the group.

Speaking in other words (or, more generally), Figure 5 exhibits snapshots taken when performing subsequent rotations with regard to the same rotation plane Π_{rot}. Firstly we have obtained the fivnum exhibited in top left of Figure 5. The initial projections of the selected ten points are represented by the filled squares, and the corresponding to them ellipse, denoted by 'a', has an extra thickness (is enhanced). Next we have continued by performing four additional rotations, using the same rotation reference plane Π_{rot}. In such way we have obtained subsequent projections of the points (shown as open circles) and subsequent ellipses. The last ellipse in the first fivnum is denoted by 'b'.

In turn we have cleared the entire plot, except the last ellipse 'b' and except the points from the last projection. The remaining now ellipse ('b') was given an extra width, and the remaining point projections were highlighted and changed their shapes to filled squares. This configuration was the starting position for the next fivnum.

In such way we have obtained the six fivnums visible in Figure 5. Each fivnum is characterized by the first and last ellipse denoted as shown in the table below:

Fivnum	Rotation	First ellipse	Last ellipse
I	$0^o \longrightarrow 60^o$	'a'	'b'
II	$60^o \longrightarrow 120^o$	'b'	'c'
III	$120^o \longrightarrow 180^o$	'c'	'd'
IV	$180^o \longrightarrow 240^o$	'd'	'e'
V	$240^o \longrightarrow 300^o$	'e'	'f'
VI	$300^o \longrightarrow 360^o$	'f'	'a'

Looking at the fivnums shown in Figure 5 one may see, that with subsequent rotations (by the angle $\alpha_1 = 15^o$) the smoothness of the transition, although not perfect, is already nearly achieved.

6 The distribution of the projected points when observed in various sectors of the projection plane

In this section we will be concerned with the distribution of the projected points (called in the following *point projections*) in the projection plane $\Pi_{proj} = <\tilde{x}_1, \tilde{x}_2>$.

Suppose, we are performing subsequent projections. For each projection we establish a random reference plane Π_{rot} for the rotation, and next the rotation angle α. The way of generating the reference planes Π_{rot} (the geodetic paths $\overrightarrow{UV}$, see Figure 1) ensures that the generated paths $\overrightarrow{UV}$ are distributed uniformly over the hypersphere. However, what can be said on the distribution of point projections observed in the projection plane $<\tilde{x}_1, \tilde{x}_2>$ in a long sequence of projections?

We have already made the remark, that the rotation angle α may be chosen in different ways. Now we will ask whether the distribution of the point projections (observed in the projection plane $<\tilde{x}_1, \tilde{x}_2>$) depends from the way of choosing the value of α, i.e. from the angle of rotation.

We firstly introduce the methodology for investigating the distribution of the projections. Next we report some of our results obtained when carrying out some longer series (*runs*) of subsequent rotations and projections.

6.1 Methodology for watching the projections

6.1.1 Introductory definitions and denotations

In this subsection we will be concerned with the following problem: When performing a longer sequence of subsequent rotations of a chosen point P, where will the point projections appear in the projection plane and which parts of that plane will they cover?

We will fix our attention on a single, arbitrary chosen point P, and next vary the chosen point.

The position (location) of a data point P in a plane can be represented in polar coordinates γ, ρ, with γ denoting the angle, and ρ denoting the radius of the polar representation. Making a sequence of s rotations and connected with them projections, we will observe in the projection plane Π_{proj} a sequence of points $P_1, P_2, \ldots, P_s$, whose coordinates will be identified by the pairs

$$(\gamma_1, \rho_1),\ (\gamma_2, \rho_2),\ \ldots\ , (\gamma_s, \rho_s).$$

Suppose, that the projection plane Π_{proj} is divided into 24 sectors centered at 0, as shown in Figure 6 (we have carried out the analysis also when considering subdivision of the projection plane into 72 sectors – the analysis did not gave any new results and is not reported here).

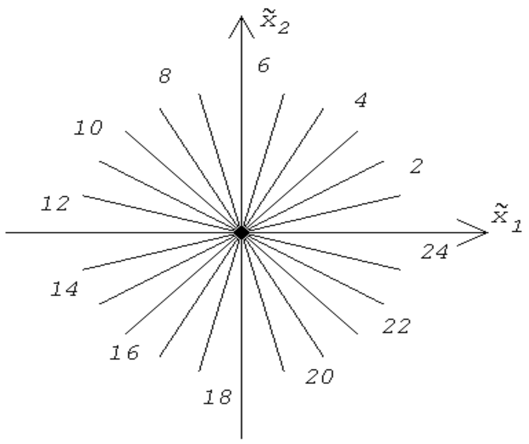

Figure 6. Design of sectors no. 1–24 in the projection plane.

Our conjecture is that – since we expect to look at the point P from nearly all 'sides' of the data cloud, its projections should appear – during a long run of the algorithm – in different sectors of the projection plane, and no sector should be granted a privilege – thus the distribution of the variables $\gamma_1, \gamma_2, \ldots, \gamma_s$ should be uniform over the sectors.

What concerns the distribution of the radii $\rho_1, \rho_2, \ldots, \rho_s$, we suppose that it will depend on the distance of the watched point P from the center of the coordinate system (being at the same time the center of the hypersphere, see Figure 1). So the distribution of the radii may depend on the shape of the data cloud and therefore be a complicated one. We do not consider it here.

In the following we will consider the distribution of the angles $\gamma_1, \ldots, \gamma_s$. We will do it by notifying – for each of the points belonging to the data cloud dealt with – the frequencies $n_1, \ldots, n_{24}$ of appearing in subsequent 24 sectors. Obviously $\sum_{i=1}^{24} n_i = s$, the total number of projections carried out.

When recording the frequencies, we have to account the following factors (conditions, circumstances), each having several levels:

1. *Location* of the sector. We will consider 24 sectors indexed as $l = 1, \ldots, 24$.
2. The *no. i* of the watched data point. We will consider each point separately. There are $n = 31$ data points in the watched set, we will index them as $i = 1, \ldots, n$.

3. The *number of projections* carried out in one experiment (run). This number will be denoted by the index s. It will take values from the set $s \in \{120, 360, 960, 1920\}$.
4. The rotation angle α, indexed as $\alpha \in \{\alpha_1, \alpha_2, \ldots\}$, where α_1 is taken equal to the angle $\measuredangle\ UOV$, and $\alpha_2, \ldots$ as some angles from the interval $[0, 2\pi)$, generated randomly in a special way, described later.

So truly the gathered frequencies are indexed by fourfold indices and each frequency of appearing in one sector can be denoted as

$$n_{l,i,s,\alpha}$$

with meaning of the indices l, i, s, α as explained above.

Additionally, for each combination of the factor levels we made several repetitions of the experiment, using the same parameters, i.e. we repeated the simulation with each of the data points (*index* i), the chosen angle of rotation (*index* α) and the number of projections (*index* s). For each combination we obtained one additional series of frequencies observed in the sectors $l = 1, \ldots, 24$.

The experiment yielded an enormous output of results, impossible to analyze 'by eye'. Thus there was a must to analyze the data by statistical methods – in order to extract the essential pattern of the results.

For our analysis of the results we have used three statistical techniques, namely the χ^2 *test*, the *indexplot of frequencies* and the *quantile–quantile plot*. Now we describe briefly each of these techniques.

6.1.2 Test χ^2 as a test for uniformity of frequencies

We know that for fixed factor levels i, s, α we obtain the series $n_{1,i,s,\alpha}, \ldots, n_{24,i,s,\alpha}$, denoted hereafter for brevity as the series of frequencies $n_1, \ldots, n_{24}$.

Suppose, the observed frequencies $n_1, \ldots, n_{24}$ are realizations of a discrete random variable Z, uniformly distributed over the set $\{1, 2, \ldots, 24\}$ with expected value $s/24$ over each value of the set.

To verify the hypothesis that the observed frequencies follow the uniform distribution specify above, we might do it by using the χ^2 test based on the following X^2 test statistics:

$$X^2 = \sum_{l=1}^{24} \frac{[n_l - E(n_l)]^2}{E(n_l)}, \qquad \text{with} \qquad E(n_l) = \frac{s}{24}. \tag{5}$$

If the distribution of the considered variable Z were uniform with the specifications just given above, then the test statistics X^2 should have a χ^2_{23} distribution (the subscript 23 denotes df, the number of degrees of freedom of the evaluated X^2 statistics; in our case $df = 23$).

The upper 0.05, 0.01 and 0.005 quantiles for the χ^2_{23} distribution amount appropriately:

$$\chi^2_{23;0.05} = 35.17, \quad \chi^2_{23;0.01} = 41.64, \quad \chi^2_{23;0.005} = 44.18. \tag{6}$$

In the following we will base our inference on the 0.01 quantile of the χ^2_{23} distribution. Thus, after obtaining a value of $X^2 > 41.64$ we will infer, that the assumption on the uniformity of appearing in the 24 sectors of the projection plane Π_{proj} is not justified and should be rejected.

6.1.3 Indexplot of frequencies

This is done to consider in more detail which sectors were visited more and which less frequently. For each sector we draw a scatterplot based on the pairs $\{l, n_l\}$, $l = 1, \ldots, 24$ (in the XlispStat numeration this will be for $l = 0, \ldots, 23$).

To show more clearly deviations from uniformity (which is our concern), we superimpose on the indexplots the value $E(n_l)$, i.e. the expected frequency evaluated under the assumption, that the hypothesis on the uniformity – of the distribution of the observed frequencies – does hold.

6.1.4 Quantile–quantile plot

The testing of uniformity may be carried out for each of the observed data points. Thus, considering n data points, we obtain n statistics X_i^2, $(i = 1, \ldots, n)$, each of which should have the χ^2_{23} distribution. A graphical summary of the obtained X^2 values may be obtained by displaying them in the form of a Q-Q (i.e. quantile–quantile) plot [6].

Q-Q plot is a technique which helps to judge by eye whether a observed series denoted generally as $y_1, \ldots, y_n$ may be considered as sampled from the assumed probability distribution $f(y; \boldsymbol{\theta})$.

We verify this conjecture by constructing two sequences

$$y_{(1)}, \ldots, y_{(n)} \quad \text{and} \quad q_1, \ldots, q_n,$$

where $y_{(i)}$ is the ith order statistics (ordered value) from the observed sequence $y_1, \ldots, y_n$, and q_i is the corresponding theoretical $(i/(n+1))$th quantile from the assumed probability distribution $f(y; \boldsymbol{\theta})$. To obtain the Q-Q plot we draw a scatterplot of the points $(q_i, y_{(i)})$, $\; i = 1, \ldots, n$.

If the points $y_1, \ldots, y_n$ were sampled from the assumed $f(y; \boldsymbol{\theta})$ distribution, then the scatterplot should exhibit points located along a straight line.

In our case we substitute for $y_1, \ldots, y_n$ the series $X_1^2, \ldots, X_n^2$, and for $f(y; \boldsymbol{\theta})$ the χ^2_{23} distribution.

6.2 Watching frequencies in 24 sectors of the projection plane

The experiment reported in this subsection has been carried out considering the farm data described in Section 3 (31 farms with $p = 5$ features each).

The simulations were done taking into account the factors (no. of sector, no. of point, number of projections, fineness (magnitude) of the rotation angle) described in subsection 6.1.

We will describe firstly the results when making rotations by the angle $\measuredangle\, UOV$, i.e. the angle between the vectors $\mathbf{u} = (u_1, \ldots, u_p)$ and $\mathbf{v} = (v_1, \ldots, v_p)$, both vectors starting from the center O and with endpoints at U, V – the randomly generated points lying on the hypersphere (see Figure 1).

Next we will report briefly the results when using for rotation an angle α established in a different way.

Table 1. Test statistics X_i^2 and selected frequencies recorded when performing rotations by the angle $\measuredangle\, UOV$

(a). Values of the test statistics X_i^2 evaluated for $n = 31$ data points recorded after performing $s = 120$, 360, 960 and 1920 projections.
Values exceeding the 0.01 quantile of the χ^2_{23} distribution are written in bold.
The framed values were obtained from frequencies shown in part (b) of the table

Nb. of projections $s = 120$ Expected frequency in one sector = 5.00									
27.60	19.60	32.80	21.20	28.80	28.40	30.80	24.80	20.40	18.80
18.40	36.00	38.40	22.00	28.80	32.40	25.60	21.20	25.60	17.60
33.20	24.00	26.40	29.20	33.60	21.20	**43.60**	26.80	$\boxed{22.80}$	$\boxed{19.20}$
$\boxed{36.00}$									
Nb. of projections $s = 360$ Expected frequency in one sector = 15.00									
29.60	30.80	30.67	23.47	33.87	35.20	23.87	36.53	36.27	24.40
29.47	**44.27**	18.27	34.27	23.47	12.53	22.27	30.93	41.33	19.73
27.20	39.60	33.60	28.40	17.60	27.07	30.00	32.67	21.87	34.53
40.80									
Nb. of projections $s = 960$ Expected frequency in one sector = 40.00									
45.20	28.65	26.65	32.95	35.40	26.65	36.60	29.25	19.85	37.10
33.75	33.45	17.90	21.10	19.00	25.10	26.05	23.40	41.30	35.45
25.00	39.85	32.05	25.30	14.85	24.20	31.10	20.65	37.40	40.65
38.05									
Nb. of rotations $s = 1920$ Expected frequency in one sector = 80.00									
55.02	36.83	38.78	23.65	41.68	38.17	35.85	40.10	30.62	38.35
47.30	41.13	19.72	25.75	25.05	34.60	30.92	30.65	**41.65**	37.45
45.80	**44.05**	38.25	**47.58**	17.00	18.30	29.02	28.70	$\boxed{\mathbf{49.90}}$	$\boxed{\mathbf{61.73}}$
$\boxed{21.27}$									

(b) Frequencies of appearing of points no. 29, 30 and 31 in subsequent 24 sectors – notified after $s = 120$ and $s = 1920$ rotations (direct output from the computer)

```
=============================================================
Nb of rotations= 120     Expected frequency in 1 sector =  5.00
no. 29   4   4   5   8   2   6   3   7   3   6   2   6
         4   5  10   5   1   2   7   8   6   7   4   5   X^2 = 22.80
no. 30   7   4   5   6   9   4   2   7   5   2   1   5
         5   6   4   3   5   4   3   4   7   8   6   8   X^2 = 19.20
no. 31   1   4   7   2   4   3  13   5   3   4   7   4
         5   8   7   5   4   4   3   0   4   8   6   9   X^2 = 36.00
=============================================================
Nb of rotations= 1920     Expected frequency in 1 sector =  80.00
no. 29  79  87  73  75  64  81  80  74  67  85  60  73
        86  78  80  91  77  66 122  94  91  95  64  78   X^2 = 49.90***
no. 30  74  67  91  78  92  74  67  96  63  55  78  56
       100  86  67  74  95  78 103  97 100  92  66  71   X^2 = 61.73***
no. 31  73  83  77  74  77  91  79  73  71  77  76  76
        87  79 103  74  76  85  80  61  82  84  88  94   X^2 = 21.23
=============================================================
```

6.2.1 Rotations using the angle $\measuredangle\, UOV$

The values X_i^2 evaluated *for all the 31 points* of the farm data, obtained when carrying out the test after $s = 120,\ 360,\ 960$ and 1920 projections are shown in part (a) of Table 1. Each value X_i^2 was obtained from a set of frequencies gathered for one (the ith) point of data – when carrying out a fixed number of s rotations.

An exemplary output of the frequencies $n_1, \ldots, n_{24}$ observed *for points no. 29, 30 and 31* is shown in part (b) of Table 1 (the calculated value of the X_i^2 statistics is denoted there as `X^2`).

Table 1 gives a flavor how various and diversiform the results may be.

Each value X_i^2 shown in part (a) of Table 1 is the value of the X^2 test statistics defined in formula (5) – calculated for a fixed combination of the investigated factors described in Section 6.1. This means that each value X_i^2 was obtained from a series of observed frequencies $n_1, \ldots, n_{24}$ gathered during one run of our experiment – for the point P_i and a fixed combination (s, α) of the last two factors. Each value X_i^2 tests the hypothesis on the uniformity of the frequencies $n_1, \ldots, n_{24}$ observed in these circumstances.

Analysing the series of the X_i^2 values, $i = 1, \ldots, 31$

We may test individually the statistical significance of each of the calculated X_i^2 values by comparing them with the appropriate critical values shown in equation [6]. Values of X_i^2 statistically significant at the level $\alpha = 0.01$, i.e. exceeding the value $\chi^2_{23;0.01} = 41.64$, are in Table 1 enhanced (written in bold).

One can see, that for $s \in \{120, 360, 960\}$ we got one significant value for each s. This eventually does not contradicts strongly the hypothesis of the assumed uniformity of the respective frequencies $n_1, \ldots, n_{24}$.

However, for $s = 1920$ we got for 8 points significant values of the test statistics X_i^2 – these were evaluated from the frequency series observed for the points $i \in \{1, 11, 19, 21, 22, 24, 29, 30\}$ – which indicates, that in some longer runs (i.e. with larger s) we obtain sometimes non–uniform distributions of the frequencies $n_1, \ldots, n_{24}$.

We may perform also a simultaneous verification whether the calculated values $X_1^2, \ldots, X_{31}^2$ follow the χ^2_{23} distribution – by constructing from them a Q-Q plot.

The respective Q-Q plots for $s = 120, 360, 1920$ are shown in Figure 7, upper plots.

Looking at the Q-Q plots displayed in the first row of figure 7 one can see that the displayed points exhibit quite a linear pattern, which indicates that the assumption on the uniform distribution may be sustained.

For comparison we have drawn 3 additional plots obtained from artificial data: the 31 values being the basis for constructing of each of the additional plots were taken as pseudo–random numbers $y_1, \ldots, y_{31}$ generated from the χ^2_{23} distribution (the values were obtained using the function `chisq-quant` from XLisp-Stat [2]).

Comparing the three plots from the upper and lower row in Figure 7 one may see that they look very much alike, and it is not possible to notice any striking difference among them.

All the plots exhibit a linear pattern. Looking at the calculated values of the X^2 statistics – obtained after performed $s = 120$ rotations – one can say, that they do not indicate any departure from uniformity. However, it seems, that in the upper right plot of Figure 7. (for $s = 1920$) the displayed values X_i^2 are somehow too large

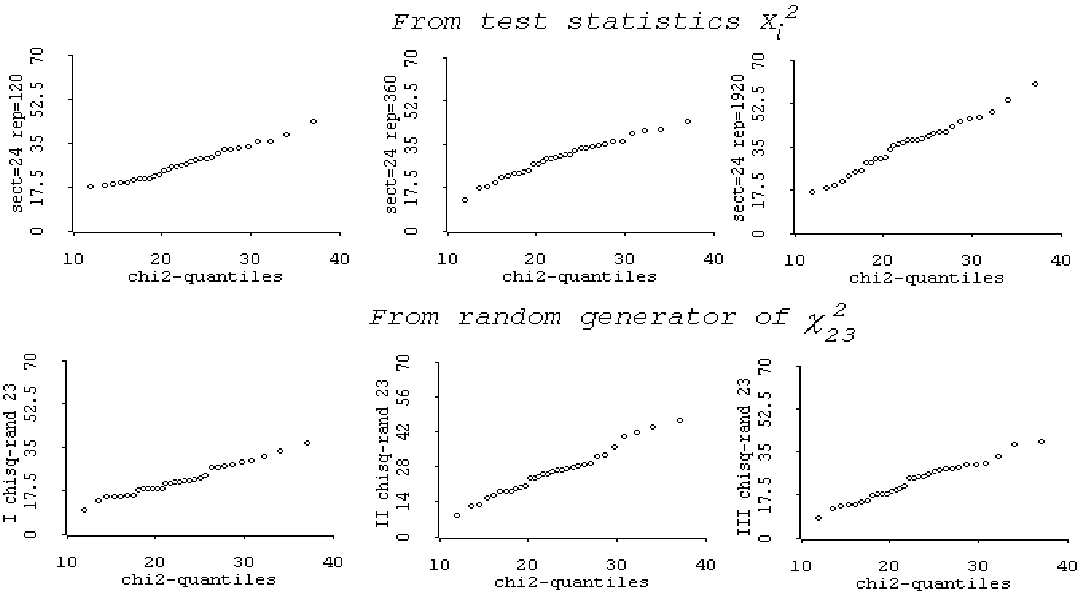

Figure 7. Q-Q plots exhibiting values of the statistics X_i^2, ($i = 1, \ldots, 31$) considered as independent realizations of a χ^2_{23} variable. Upper row shows Q-Q plots constructed from calculated X_i^2 values evaluated from frequencies gathered after carrying out s rotations by the angle $\measuredangle\, UOV$; each point in the plot represents a X_i^2 value calculated for one data point by notifying its appearances in 24 sectors of the projection plane. Bottom row shows analogous Q-Q plots constructed from pseudo-random numbers $y_1, \ldots, y_{31}$ generated from the χ^2_{23} distribution

as compared to those obtained from the χ^2_{23} distribution. This happened in some other runs too. We have observed that for some combinations we got too many significant X_i^2 values of the X^2 statistics. We could not find any systematic pattern of this happening.

Looking specifically at the frequencies of points no. 29, 30, and 31

It is impossible to look at all the gathered frequencies, thus we have fixed our attention specially on points no. 29, no. 30 and no. 31. The point no. 30 is located at the periphery of the data cloud and was notified as outlier (see Figure 2, where it appears as point #29, because the numeration in XLispStat starts from 0).

There seems to be nothing special about the position of points no. 29 and no. 31. The individual frequencies gathered for these points after carrying out $s = 120$ and $s = 1920$ rotations are shown in Table 1, bottom part. Because it is much easier to judge graphically (by eye) the uniformity of the distribution of the frequencies, we have displayed the results obtained for the selected data points in the form of indexplots with the index focusing on sectors. The respective indexplots – of the frequencies recorded after $s = 120, 360, 960$ and 1920 rotations – are shown in Figure 8, in which also the respective expected values of the frequencies (expected under assumption of uniformity) are superimposed.

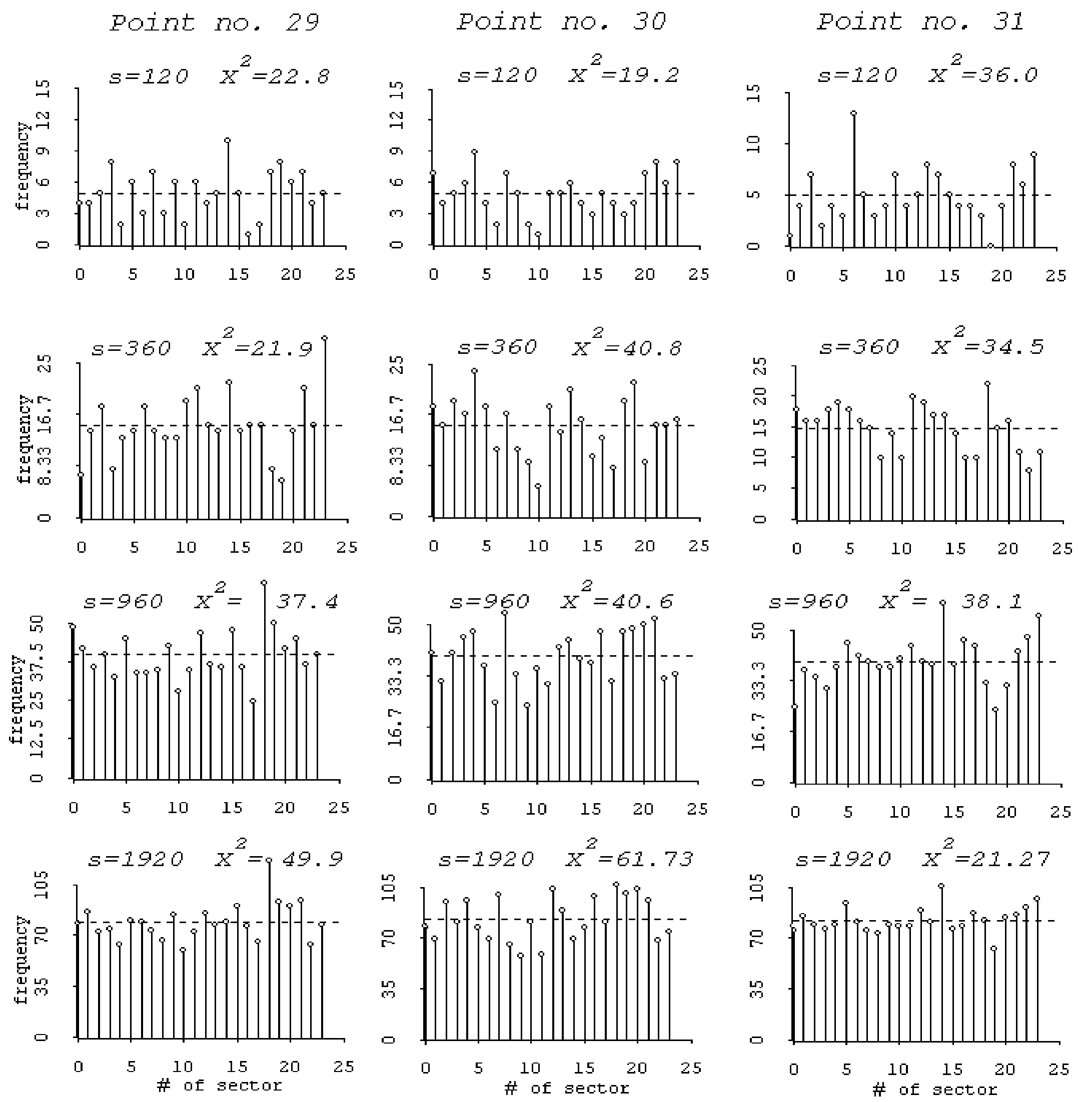

Figure 8. Indexplots displaying the frequencies of appearing of points no. 29, 30 and 31 in 24 sectors of the $<x_1, x_2>$ *plane, when making rotations by the angle* $\alpha = \measuredangle\, UOV$*. The expected frequency of appearing – under assumption of the uniform distribution – is marked by the dashed line. Values of* $X^2 > 41.64$ *are statistically significant at the* $\alpha = 0.01$ *level, which means that they indicate for a departure from the uniform distribution in sectors*

One can see, that the uniformity of appearing of point projections in the sectors $l = 1, \ldots, 24$ (labelled in the plots as $\tilde{l} = 0, \ldots, 23$) is not very bad. Already after $s = 120$ projections each of the sectors was visited at least once (except for the sector #19, point no. 31).

However, the distribution remains scattered, especially for larger number of s, and we note some significant values, indicating for a departure from uniformity.

In other runs we got similar results – without any systematic pattern. We were not able to link the observed discrepancies (big X^2 values) neither with individual data points nor with the number of performed rotations.

We can not explain the fact fully at this moment. We feel that this might be connected with the fact that, despite all, the positions of the observed projections are not completely independent. For a given point, its position after rotation (especially when the rotation angle is small) may depend from its position before the rotation. Thus, in some runs some spurious dependencies from the former position of the projection may accumulate.

6.2.2 Analysis for rotations using angles differing from $\measuredangle UOV$

To throw some light on the supposition, that the uniformity of distribution of the frequencies may depend on the choice of the angle of rotation, we have performed a series of experiments, in which we have declared directly the rotation angle α.

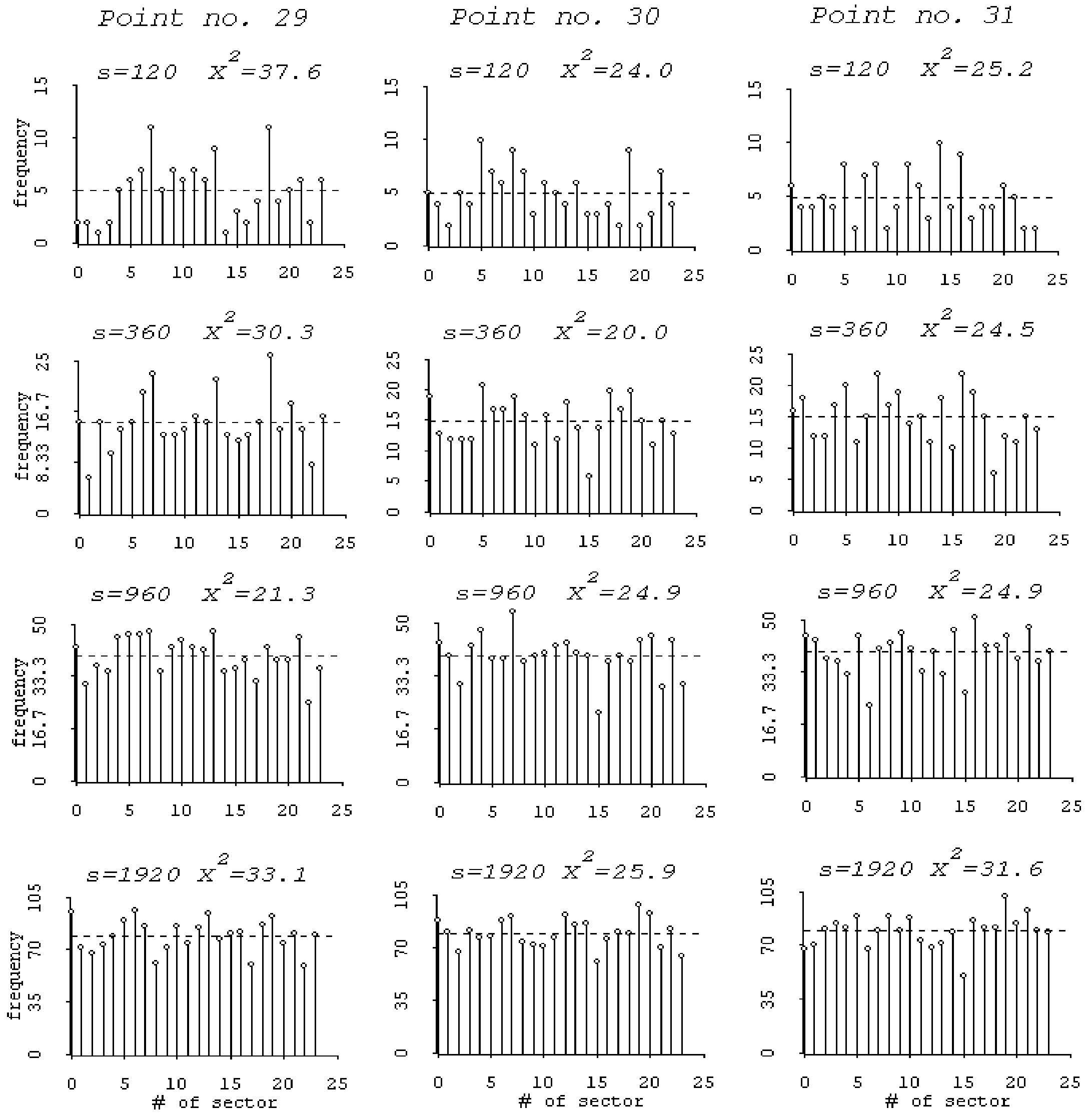

Figure 9. Indexplots displaying the frequencies of appearing of points no. 29, 30 and 31 in 24 sectors of the $<x_1, x_2>$ plane, when making rotations by the angle $\alpha = [126 + random(108)]^o$. The expected frequency of appearing – under assumption of the uniform distribution – is marked by the dashed line

We notified without doubt, that when choosing smaller angles (e.g. 30^o, 45^o, 60^o exactly, or randomly generated with expected value in that range) the evaluated values of the X^2 statistics were decidedly elevated for all points, thus indicating for non uniformity of the distribution of the frequencies.

We have performed also a series of experiments, choosing the angle of rotation from the interval: $[126+random(120)]^o$. Applying rotation angles approaching 180^o we got in many runs (but not in all) nice uniform distributions. Indexplots for such series, when the rotation angle was generated from the interval $[126+random(108)]^o$ are shown in Figure 9.

Generally, one could say that the plots displayed in Figure 9 do not differ much from those, displayed in Figure 8.

However, for larger values of s we find in Figure 8 more significant values of the statistics X^2 indicating for a departure from uniformity of the distribution.

This is gathered in the table below:

	$\alpha = \measuredangle UOV$			$\alpha = [126 + random(108)]^0$		
	$no.29$	$no.30$	$no.31$	$no.29$	$no.30$	$no.31$
$s = 120$	22.8	19.2	36.6*	37.6*	24.0	25.2
$s = 120$	21.9	40.8*	34.5	30.3	20.0	24.5
$s = 120$	37.4*	40.6*	38.1*	21.3	24.9	24.9
$s = 120$	49.9***	61.7***	21.3	33.1	25.9	31.6

Critical values: $\chi^2_{0.05} = 35.17$, $\chi^2_{0.01} = 41.64$, $\chi^2_{0.005} = 44.18$

7 Conclusions

After all the investigations concerning the angle of the rotation we came to the conclusion, that the angle could be taken either as the angle $\measuredangle\ UOV$ designated by the randomly generated points on the hypersphere (1st proposal) or an angle generated randomly from the interval $[126 + random(108)]^o$ (2nd proposal). This result was obtained when investigating the uniformity of the point projections in the projection plane.

Repeating the investigations in several series of experiments, we observed that in the long run – when using α according to second proposal, we have obtained more frequently a perfect fit with the uniform distribution, as that happened when using the first proposal.

Thus, in our implementation of the grand tour, we have introduced the second proposal. We advise to make the (global) rotation by the angle $\alpha = [126 + random(108)]^o$, composing α from small angles α_1, say $\alpha_1 = 9^o$ or 15^o.

Acknowledgements

The work was sponsored partially from the KBN grant no. 8T11C 04011.

The authors are grateful to anonymous referees for their careful reading of the first version of the paper; their comments made appear the paper more concise and comprehensive.

References

1985

[1] Asimov D.: The grand tour: a tool for viewing multidimensional data. *SIAM J. Sci. Stat. Comput.* 6, 1, 128-143.

1990

[2] Tierney L.: *LISP–STAT, An Object–Oriented Environment for Statistical Computing and Dynamic Graphics*, Wiley, N.Y.

1994

[3] Swayne D.F., Cook D., Buja A.,: Interactive dynamic graphics in the X Windows system with a link to S. In: *ASA Proceedings of the Section on Statistical Graphics*, 1–8.

1996

[4] Bartkowiak A., Szustalewicz A.: Some issues connected with a 3D representation of multivariate data points. *Machine Graphics & Vision* 5, No. 4, 563–577.

1997

[5] Bartkowiak A., Szustalewicz A.: The grand tour as a method for detecting multivariate outliers. *Machine Graphics & Vision* 6, No. 4, 487–505.

[6] Gnanadesikan R., *Methods for Statistical Data Analysis of Multivariate Observations.* Wiley, New York. 2nd Edition (1st Edition 1977).

Appendix. Derivation of the rotation matrix for the transformation $\mathbf{y} = \mathbf{x}A$

1. Two points U, V lying on the hypersphere may be interpreted as the unit length row vectors $\mathbf{u} = [u_1, \ldots, u_p]$ and $\mathbf{v} = [v_1, \ldots, v_p]$ with their begins at the center O and their end points at U and V respectively. We will assume that the angle ϕ between these vectors differs from 0^o and also from 180^o within an assumed accuracy ϵ, set at the begin of the calculations.

For further calculations it will be convenient to orthogonalize the vectors $\mathbf{u}$, $\mathbf{v}$. We do it by applying the *Gram-Schmidt orthonormalization* method. It is enough to change the vector $\mathbf{v}$ to the new one defined as

$$\mathbf{t} := \frac{\mathbf{v} - (\mathbf{u}, \mathbf{v})\mathbf{u}}{||\mathbf{v} - (\mathbf{u}, \mathbf{v})\mathbf{u}||},$$

where the scalar product $(\mathbf{u}, \mathbf{v})$ denotes *cosinus* of the angle ϕ between the vectors $\mathbf{u}$ and $\mathbf{v}$:

$$(\mathbf{u}, \mathbf{v}) = \mathbf{u}\mathbf{v}^T = \sum_{i=1}^{p} u_i v_i = \cos\phi\,.$$

(The vectors $\mathbf{u}$ and $\mathbf{t}$ are orthonormal.)

Let $<\mathbf{u}, \mathbf{t}>$ denote the 2D-plane spanned by the vectors $\mathbf{u}$ and $\mathbf{t}$.

2. Suppose, we have given a vector $\mathbf{x} \in R^p$, not collinear with the vectors $\mathbf{u}, \mathbf{t}$. We want to rotate the vector $\mathbf{x}$ by an angle α in such a way, that the endpoint X of $\mathbf{x}$ is displaced in a plane parallel to the plane $<\mathbf{u}, \mathbf{t}>$, keeping a constant distance of the (rotated) endpoint from $<\mathbf{u}, \mathbf{t}>$.

By projecting the vector $\mathbf{x}$ onto $<\mathbf{u},\mathbf{t}>$ we decompose it into two components

$$\mathbf{x} = \mathbf{d} + \mathbf{r},$$

where:

$\mathbf{d} = \overrightarrow{OD}$ is the orthogonal projection of $\mathbf{x}$ onto $<\mathbf{u},\mathbf{t}>$,

$\mathbf{r} = \overrightarrow{DX}$ is the orthogonal complement of the vector $\mathbf{d}$ (to $\mathbf{x} = \overrightarrow{OX}$) – see Figure 10 (left).

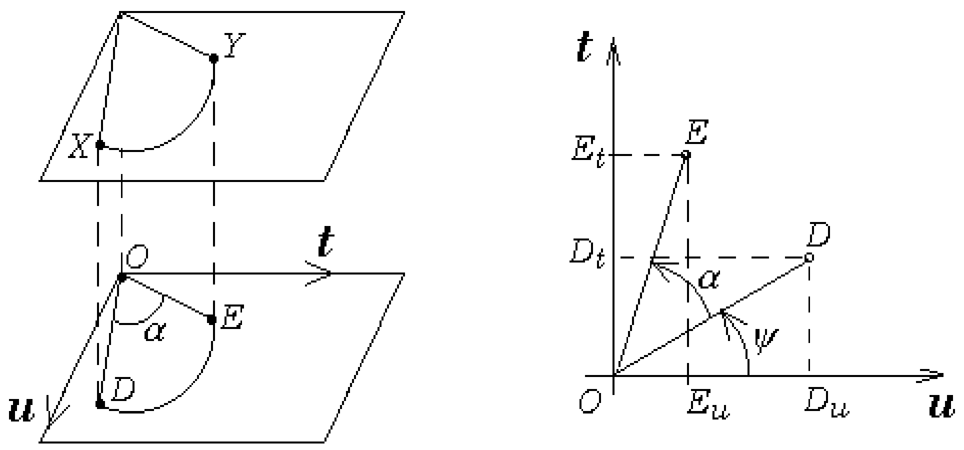

Figure 10. Rotation of the vector $\overrightarrow{OX}$ with regard to the plane $<\mathbf{u},\mathbf{t}>$.
Left: Rotating the endpoint of $\overrightarrow{OX}$ in the plane parallel to $<\mathbf{u},\mathbf{t}>$.
Right: Rotating the vector $\overrightarrow{OD}\in<\mathbf{u},\mathbf{t}>$ by the angle α to the position $\overrightarrow{OE}$.

In turn the vector $\mathbf{d}$ may be decomposed as

$$\mathbf{d} = (\mathbf{x},\mathbf{u})\mathbf{u} + (\mathbf{x},\mathbf{t})\mathbf{t}, \tag{7}$$

with $(\mathbf{x},\mathbf{u})$ and $(\mathbf{x},\mathbf{t})$ denoting the scalar products of the vectors $\mathbf{x},\mathbf{u}$ and $\mathbf{x},\mathbf{t}$ appropriately.

Then $\mathbf{r}$ may be calculated as:

$$\mathbf{r} = \mathbf{x} - \mathbf{d} = \mathbf{x} - (\mathbf{x},\mathbf{u})\mathbf{u} - (\mathbf{x},\mathbf{t})\mathbf{t}. \tag{8}$$

Now suppose, we rotate the vector $\mathbf{d} = \overrightarrow{OD}$ in the plane $<\mathbf{u},\mathbf{t}>$ by an angle α obtaining this way its new position $\mathbf{e} = \overrightarrow{OE}$. This situation is depicted in Figure 10 (right).

Let us denote by ρ the length of the vector $\mathbf{d}$. The vector $\mathbf{e}=\overrightarrow{OE}$, resulting from the rotation of $\mathbf{d}$, is obviously of the same length ($= \rho$). The position of the point E (being the endpoint of $\mathbf{e}$), is given by its coordinates (E_u, E_t), which may be calculated from the length ρ and the angles ψ and α as follows:

$$\begin{aligned} E_u = \rho\cos(\psi+\alpha) &= \rho\cos(\psi)\cos(\alpha) - \rho\sin(\psi)\sin(\alpha) \\ &= (\mathbf{x},\mathbf{u})\cos\alpha - (\mathbf{x},\mathbf{t})\sin\alpha\,, \\ E_t = \rho\sin(\psi+\alpha) &= \rho\cos(\psi)\sin(\alpha) + \rho\sin(\psi)\cos(\alpha) \\ &= (\mathbf{x},\mathbf{u})\sin\alpha + (\mathbf{x},\mathbf{t})\cos\alpha\,. \end{aligned}$$

Hence

$$\mathbf{e} = \overrightarrow{OE} = \overrightarrow{OE}_u + \overrightarrow{OE}_t = [(\mathbf{x},\mathbf{u})\cos\alpha - (\mathbf{x},\mathbf{t})\sin\alpha]\mathbf{u} + [(\mathbf{x},\mathbf{u})\sin\alpha + (\mathbf{x},\mathbf{t})\cos\alpha]\mathbf{t}\,. \quad (9)$$

In such a way we have obtained the position of the rotated vector $\mathbf{d}$ (coordinates of its new endpoint E).

3. To obtain the position of the rotated vector $\mathbf{x}$ (coordinates of its new endpoint Y attained after the displacement) we proceed as follows. Let us denote the rotated vector $\mathbf{x}$ by $\mathbf{y}$. Then obviously (see Figure 10 (left))

$$\mathbf{y} = \overrightarrow{OY} = \overrightarrow{OE} + \overrightarrow{EY} = \mathbf{e} + \mathbf{r}.$$

Taking into account the formulae (9) and (8) for $\mathbf{e}$ and $\mathbf{r}$ we may write:

$$\begin{aligned}
\mathbf{y} &= \mathbf{e} + \mathbf{r} \\
&= [(\mathbf{x},\mathbf{u})\cos\alpha - (\mathbf{x},\mathbf{t})\sin\alpha]\mathbf{u} + [(\mathbf{x},\mathbf{u})\sin\alpha + (\mathbf{x},\mathbf{t})\cos\alpha]\mathbf{t} + \mathbf{x} - (\mathbf{x},\mathbf{u})\mathbf{u} - (\mathbf{x},\mathbf{t})\mathbf{t} \\
&= \mathbf{x} + (\mathbf{x},\mathbf{u})[(\cos(\alpha)-1)\mathbf{u} + \sin(\alpha)\mathbf{t}] + (\mathbf{x},\mathbf{t})[-\sin(\alpha)\mathbf{u} + (\cos(\alpha)-1)\mathbf{t}]\,.
\end{aligned}$$

Expressing now the scalar products $(\mathbf{x},\mathbf{u})$ and $(\mathbf{x},\mathbf{t})$ in the form of multiplication:

$$(\mathbf{x},\mathbf{u}) = \mathbf{x}\mathbf{u}^T\,, \qquad (\mathbf{x},\mathbf{t}) = \mathbf{x}\mathbf{t}^T,$$

the formula for $\mathbf{y}$ may be changed to the more simple form:

$$\mathbf{y} = \mathbf{x}\mathbf{A} \quad (10)$$

where the matrix $\mathbf{A}_{p\times p}$, called also the *rotation matrix* or the *transformation matrix*, is given by the formula:

$$\mathbf{A} = \mathbf{I} + \mathbf{u}^T[(\cos(\alpha)-1)\mathbf{u} + \sin(\alpha)\mathbf{t}] + \mathbf{t}^T[-\sin(\alpha)\mathbf{u} + (\cos(\alpha)-1)\mathbf{t}]\,, \quad (11)$$

with $I_{p\times p}$ denoting the unit matrix.

It may be proved, that the derived matrix $\mathbf{A}$ is orthogonal, i.e. $\mathbf{A}^T\mathbf{A} = \mathbf{A}^T\mathbf{A} = \mathbf{I}$, and $\mathbf{A}^{-1} = \mathbf{A}^T$.

It can be seen, that transformation by use of the matrix $\mathbf{A}$ as defined above preserves the length of any vector $\mathbf{x}$, to which it is applied. Let $\mathbf{y}$ denote the transformed vector $\mathbf{x}$. Then:

$$||\mathbf{y}||^2 = \mathbf{y}\mathbf{y}^T = (\mathbf{x}\mathbf{A})(\mathbf{x}\mathbf{A})^T = \mathbf{x}\mathbf{A}\mathbf{A}^T\mathbf{x}^T = \mathbf{x}\mathbf{x}^T = ||\mathbf{x}||^2.$$