Multidimensional Visualization and It's Applications

TUTORIAL – Intelligent Data Analysis 2003

ALFRED INSELBERG[1]

Departments of Computer Science & Applied Mathematics
Tel Aviv University
Ramat-Aviv, Tel Aviv 61390
Israel
aiisreal@math.tau.ac.il
&
Multidimensional Graphs Ltd[2]

Berlin
August 28, 2003

[1]Senior Fellow San Diego SuperComputing Center
[2]36A Yehuda Halevy Street, Raanana 43556, ISRAEL

Abstract

The desire to augment our 3-dimensional perception and the need to understand multivariate problems spawned several multidimensional visualization methodologies. Starting from early successes of visualization, like Dr. J. Snow's dot map in 1854 showing the connection of cholera to a water pump in London, Scatter plots, Chernoff faces, Andrews plots, Projection Pursuit, Perceptualization of data, Data density, Trees and Castles, Kinematic displays, Bertin Permutation Matrices and other multivariate techniques have been developed (see Bibliography A in the Appendix). Some of these will be reviewed in order to establish the connection between multivariate problems and multidimensional geometry. Understanding the underlying geometry of a multivariate problem provides important insights into what is possible and what is not. For the unambiguous visualization of multidimensional geometry and, in turn, multivariate relations Parallel Coordinates – the leading Multidimensional Vis Methodology – is introduced and rigorously developed. Relations among N real variables are mapped uniquely into subsets of 2-space having geometrical properties enabling the visualization of the corresponding N-dimensional hypersurfaces.
After the basic representation results, associated algorithms for constructions, intersections, transformations, containment queries, proximity and others will be presented. The development is interlaced with applications of the relevant results starting with demonstrations of Data Mining on real datasets (i.e. Feature extraction from LandSat data, Financial, Process Control, Pilot Selection, Raising the Yield and Quality of VLSI chips, and others). They are followed by Collision Avoidance Algorithms for Air Traffic Control which are based on the representation of lines in multidimensional space. The detection of coplanar points and the representation of planes and hyperplanes lead to some applications in Computer Vision, Geometric Modeling and elsewhere. More examples of Visual Data Mining are given. An efficient geometric automatic **classifier** algorithm is motivated and is demonstrated on some challenging datasets. Finally, the representation of curves and hypersurfaces is taken up together with interactive applications to Process Control, Instrumentation and Heuristic Optimization. Nonlinear VISUAL models, in terms of hypersurfaces, are constructed from data and used interactively for Decision Support, Sensitivity Analysis, studying feasibility and effect of constraints as well as trade-off analysis.
NOTE: Do not be intimidated by the formalistic language. The organizer is also well known for numerological anecdotes and palindromic diversions. Valuable prizes will be distributed in real-time to those contributing memorable and noteworthy digressions.

KEYWORDS: Multidimensional Geometry, Multidimensional/Multivariate Visualization, Information Visualization, Parallel Coordinates, Visual & Automatic Data Mining, Intelligent Process Control & Instrumentation, Nonlinear Modeling, Decision Support.

Contents

List of Figures

VISUALIZATION - An Introduction

** **Insight through Images — in the spirit of Hamming's "we compute to gain insight not numbers". Over half of our sensory neurons are devoted to vision. A goal of Visualization is to incorporate our tremendous pattern-recognition ability in our problem-solving loop.**

** **Emerging Field with Huge Potential – Propelled by Technological Advances and the need to Visualize the "Unseen".**

** **Seminal Report** *Visualization in Scientific Computing*, **ACM SIGGRAPH 1987 promoted Scientific Visualization and indirectly Visualization in other fields.**

** **Techniques are ad hoc and application specific. Roughly speaking the field consists of a collection of mappings :**

$$Problem(s)\ Class \rightarrow Visual\ Models$$

** **"Escaping flatland is the essential task of envisioning information - for all the interesting worlds (physical, biological, imaginary, human) that we seek to understand are inevitably and happily MULTIVARIATE in nature. Not flatlands." – E. R. Tufte preface in *Envisioning Information*, Graphic Press, Cheshire, Conn. 1990.**

** **Our goal is the visualization of complex problems with many parameters – *Multivariate Visualization* or equivalently *Multidimensional Visualization* we shall emphasize *Information Visualization*.**

** **Believe it or not, the fascination with Dimensionality may predate Aristotle and Ptolemy who argued that space can only have three dimensions. By the nineteenth century, mathematicians like Riemann, Lobachevsky and Gauss unshackled our imagination and higher-dimensional and non-Euclidean geometries came into their own. The intellectual challenge, limited by our 3-dimensional perceptual experience, and the abundance of multivariate problems, spawned various methodologies to represent (encode) finite sets of multivariate data points as indicated in bibliography (APPENDIX A)(It is worthwhile doing a search on WWW for "Multivariate, or Multidimensional or Information Visualization").**

** **What is needed is a** *conceptual breakthrough* **to enable the visualization not only of** *Multivariate* **Data but also of** *RELATIONS* **without Loss of Information.**

MULTIDIMENSIONAL VISUALIZATION

We focus on the leading multidimensional methodology for the visual presentation of relationships between many variables. It is based on a system of *Parallel Coordinates* **(abbr. ||-coords) and provides a a systematic and rigorous way of visualizing N-Dimensional geometry. This is in the** *Spirit of Descartes* **whose coordinate system enables us to tranform relations between 2 and 3 variables (dimensions) to geometric models – their graphs. However, rather than using** *orthogonal* **axes we place them in** *parallel* **for orthogonality "uses up" the plane very fast. It is *Parallelism* rather than orthogonality which is the fundamental concept in Geometry, and contrary to popular belief the concepts are not equivalen. A notion of *angle* is required for orthogonality whereas for parallelism what is needed are lines without points in common.**

Based on the experience acccumulated thus far the properties which a desirable multidimensional visualization methodoly should have are listed next. You are encouraged to contribute your own ideas and requirements.

WANTED!

A Multidimensional Visualization Methodology which

displays multivariate/multidimensional relations

♡ **without loss of information, and low representational complexity (i.e. for Parallel Coordinates the complexity is $O(N)$ while for the common scatterplot matrix it is $O(N^2)$),**

which works for any number of dimensions/variables,

♡ **and treats every variable in the same way,**

♡ **enables the object being displayed to be recognized under projective transformations (i.e. translations, rotations, scaling and perspective),**

♡ **such that the properties of the relation uniquely correspond to the properties of its image, and**

♡ **is based on a body of rigorous mathematical and algorithmic results (that is theorems on how certain objects are displayed rather than ad hoc heurestics).**

FORMAL OVERVIEW

This part is for the more mathematicaly inclined.

Don't let the notation intimidate you!

⋆ **A** $RELATION$ **between N real variables** $x_1, x_2, \ldots, x_N$ **is a subset** F **of** R^N **– the Euclidean N - Dimensional Space i.e.** $F \subset R^N$**.**

⋆ **In order not to lose information we want to map** F $uniquely$ **into a planar pattern i.e. a subset of** R^2 **– which is a relation between** x_1, x_2**.**

⋆ **The plan, then, is to construct a mapping**

$\mathscr{J} : 2^{P^N} \rightarrow 2^{P^2}\ X\ IndexSet$

where $2^A = \{B \mid B \subset A\}$ **is the** $power\ set\ of\ A$**, which maps subsets of** P^N**, the** $Projective$ **rather than the than Euclidean N - space into subsets of 2-space. The reasons for using the Projective space as well as the *Index Set* will be explained shortly.**

⋆ **Further,** $\mathscr{J}$ **should be** $one-to-one$ **so that** $\mathscr{J}(F_1) = \mathscr{J}(F_2) \Leftrightarrow F_1 = F_2$**. By the way, since the** $cardinality$ **of** 2^{P^N} **and** 2^{P^2} **are the same, it is in principle possible to construct an** $\mathscr{J}$ **satisfying this requirement.**

⋆ **It will be shown that it is possible to construct such a mapping,** $\mathscr{J}$**,** $recursively$

on the dimensionality of the object being represented. That is, starting with points (0-dimensional) – this the non-recursive part directly from the definition, then successively taking the envelopes of the polygonal lines (1-dimensional), p-flats (p-dimensional planes $0 \leq p \leq N-1$), then certain hypersurfaces. In this the indexing plays a crucial role.

⋆ **A subset F of P^N is then *represented* by its image $\bar{F} = \mathscr{J}(F)$. We would like $\bar{F}$ to have geometrical properties which will aid our intuition to discover the properties of the N-dimensional subset F that it represents. This, of course, is a cognitive and subjective requirement.**

All this formalism will be clarified in the ensuing. There are a couple of misconceptions, however, that are worth clearing up at this stage.

Occassionaly, mappings between N-space and M-space, where $N > M$, are erroneously referred to as *projections*; but not all such mappings are *projections*. Specifically, a projection from N to M space takes a point $P(x_1, x_2, ..., x_M, x_{(M+1)}, ..., x_N)$ into the point $P'(x_1, x_2, ..., x_M)$. Hence it only has information about the M variables it retains. So for our purposes, projections are not desirable since they lose *all* information about the N-M missing variables. Here $\mathscr{J}$ is not a projection and, in fact, is not even a point-to-point mapping. This is par-

ticularly relevant to *scatterplots* which are a very important technique commonly used in multivariate visualization. When the number of variables is N, a scatterplot matrix consists of the $N(N-1)/2$ projections of the N variables taken pairwise. Unfortunately, even such a plethora of projections may lose information about the N-dimensional object it portrays. In 3-D for example, consider the symmetric intersection of 3 cylinders having the same radius r. The 3 pairwise projections of this object are identical to those of a sphere with radius r, and hence these two relatively simple objects can not be distinguished by their projections. It is worth coming up with your own examples.

Also don't let the name fool you, the *Projective Plane*, which we will be occassionally mentioning is not related to projective mappings.

PARALLEL COORDINATES – Definition

In the Euclidean plane with *xy*-Cartesian coordinates, N copies of the real line labeled $\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_N$ are placed equidistant (e.g. one unit apart) and perpendicular to the *x*-axis. They are the axes of the parallel coordinate system for N-space all having the same positive orientation as the *y*-axis. A point C with coordinates $(c_1, c_2, \ldots, c_N)$ is represented by the complete polygonal line $\bar{C}$ (i.e. the lines of

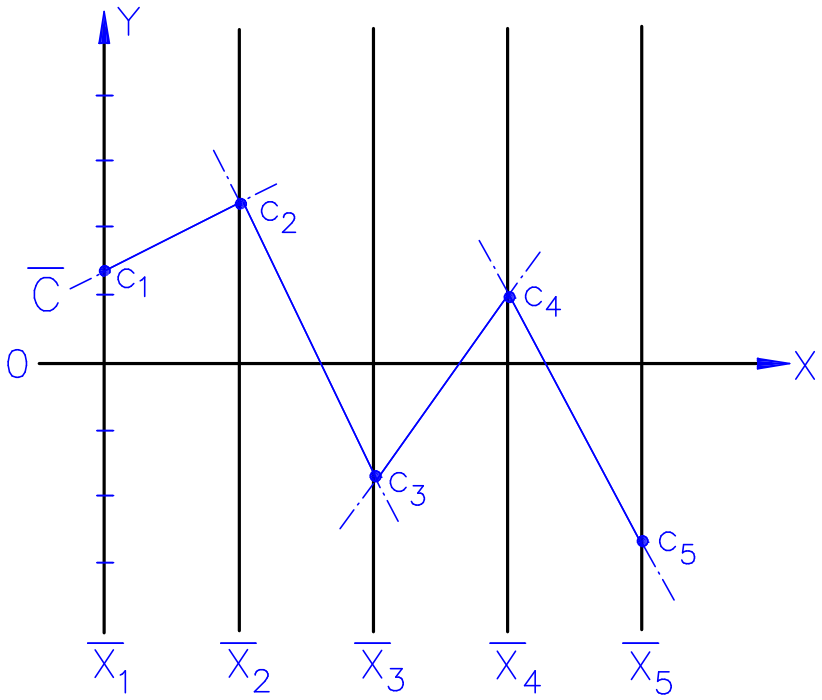

Figure 1: The polygonal line $\overline{C}$ represents the point $(c_1, c_2, c_3, c_4, c_5)$.

which only the local segments are usually shown) whose N vertices are at $(i-1, c_i)$ on the $\overline{X_i}$-axis for $i = 1, \ldots, N$ as shown in Fig. 1. In this way, a 1-1 correspondence between points in N-space and planar polygonal lines with vertices on the parallel axes is established. The definition is deceptively simple and many people stop here without realizing the power of Parallel Coordinates which is really a whole METHODOLOGY.

In Fig. 2 , $N = 7$ with $r = r_7$ being the required distance. One of the points shown is the origin though the construction is valid in general.

Here there will be a static display of a multivariate dataset with 35 parameters and thousands of data items. Then a set of LandSat data will be examined showing

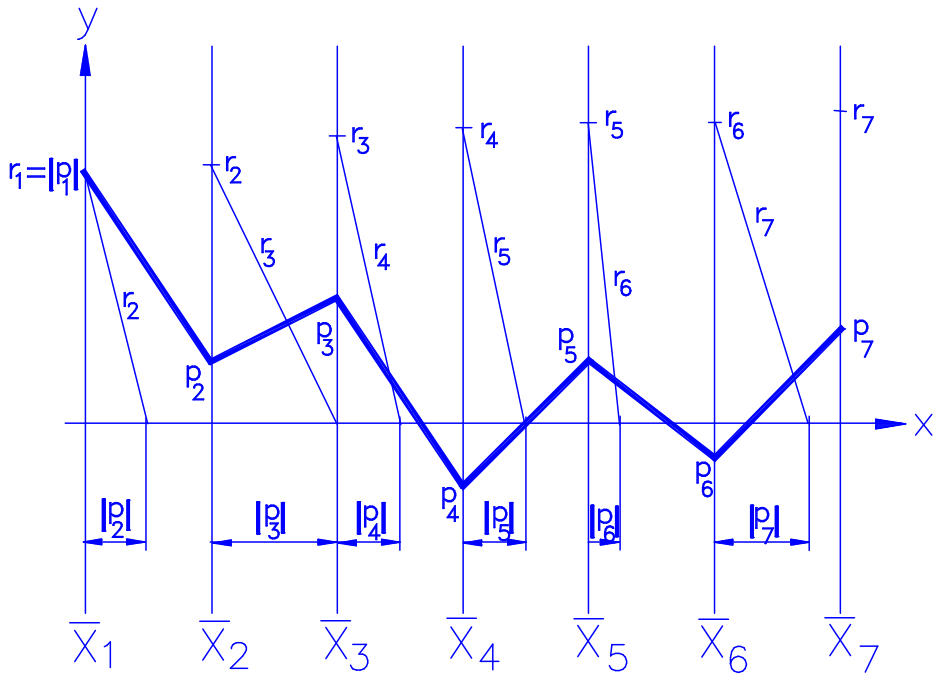

Figure 2: Constructing the Euclidean Distance between two points.

how feature extraction can be accomplished interactively.

REPRESENTING RELATIONS – START WITH 2-D

What distinguishes ||-coords from Nomography, "Profiles", "Glyphs", "N-M plots", "Andrews Curves", "Chernoff's faces", etc is the ability to represent and display not only points but also multivariate *relations* without losing information. We start our exploration in 2-D not only because it is the simplest, but also because we can contrast ||-coords with Cartesian coordinates.

As we see from Fig. 3 a *point* is represented by a line *line*. And it is, therefore, natural to ask "how is a line represented"?

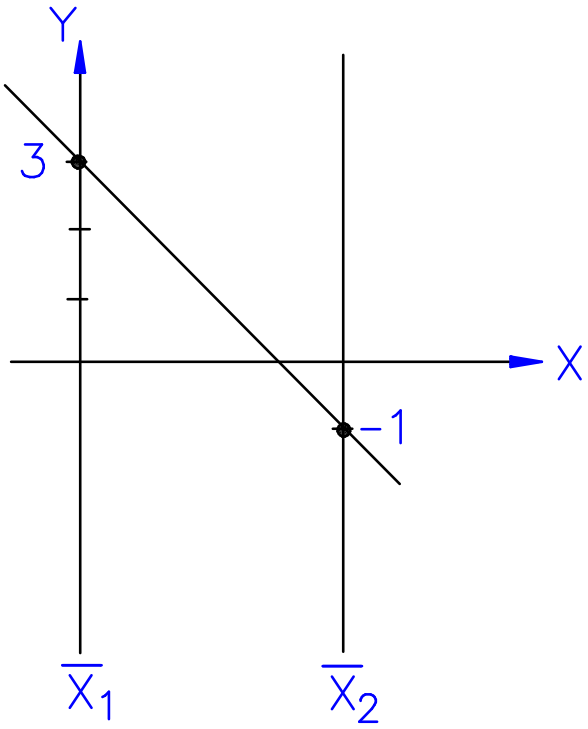

Figure 3: A Point, (3, -1), in 2-D is represented by a line

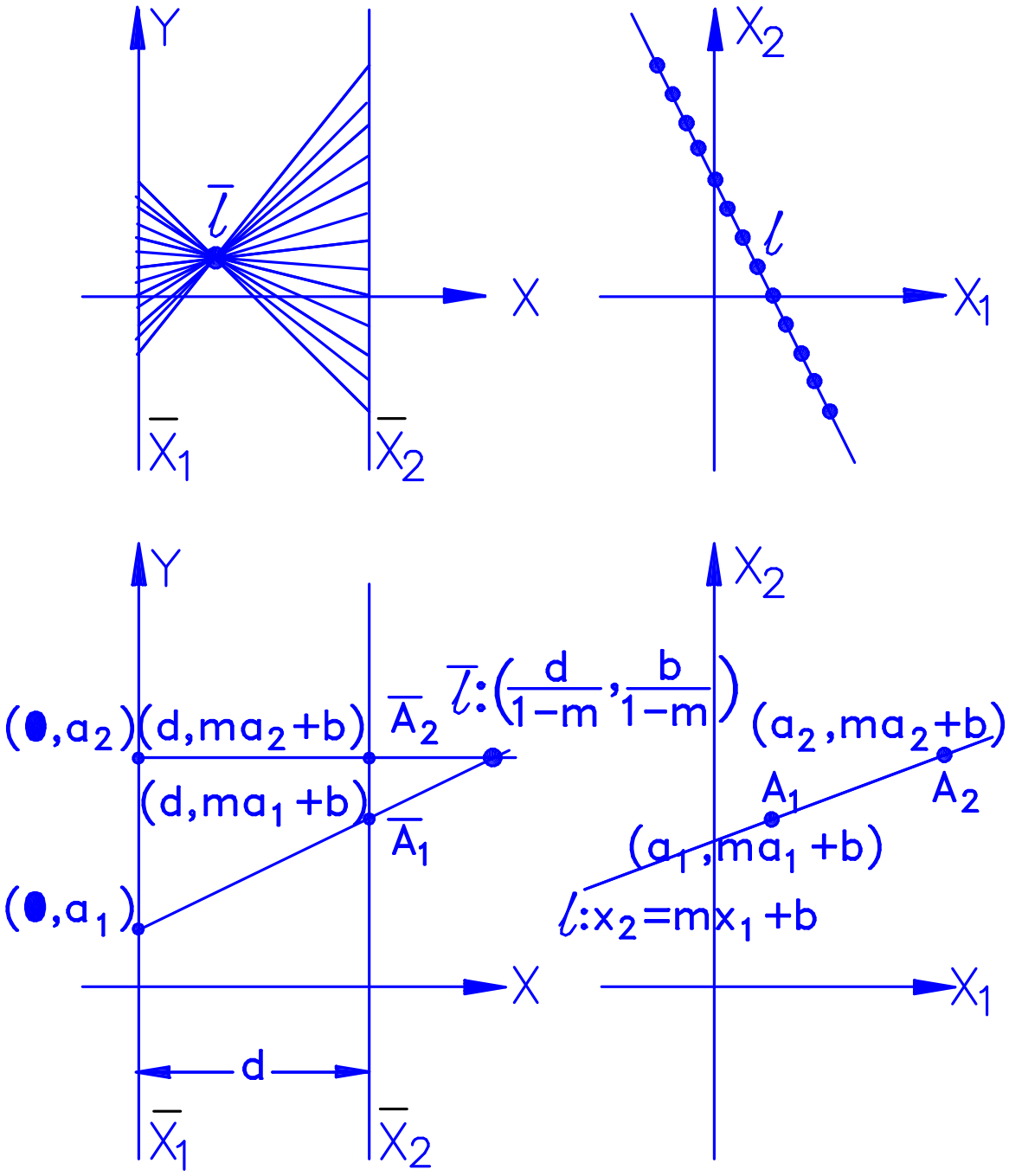

Figure 4: Point $\longleftrightarrow$ Line duality in 2-D

In Fig. 4, the distance between the parallel axes is d. The line

$$l : x_2 = mx_1 + b, \tag{1}$$

is a collection of its points A. In turn, the points are represented in $\|$-coords by the infinite collection of lines $\bar{A}$ on the xy plane. Remarkably, when $m \neq 1$ these lines intersect at the *point* with xy-coordinates:

$$\bar{l} : (\frac{d}{1-m}, \frac{b}{1-m}). \tag{2}$$

This motivates the tentative (which will be modified as we go along – see Recursive Definition in the previous section) definition on the representation of relations. Namely, a relation, typically involving infinitely many points, will be represented by the *envelope* of the corresponding infinite family of polygonal lines representing the points of the relation.

The *point* (2) represents the linear *relation*, Equation (1), and is, in fact, the envelope of the family of lines $\bar{A}$. This point, all by itself, suffices to represent the line for the two parameters m and b specify completely both l and $\bar{l}$. In effect, $\|$-coords in 2-D induce a *Point* $\rightleftharpoons$ *Line* duality (i.e. mapping points into lines and vice versa – this is examined more thoroughly in the next section). But there is a "little problem" when $m = 1$. Despite appearances, the point $\bar{l}$ does not "blow up" as $m \to 1$. Rather, when the limiting process is done carefully, one sees that $\bar{l}$ goes

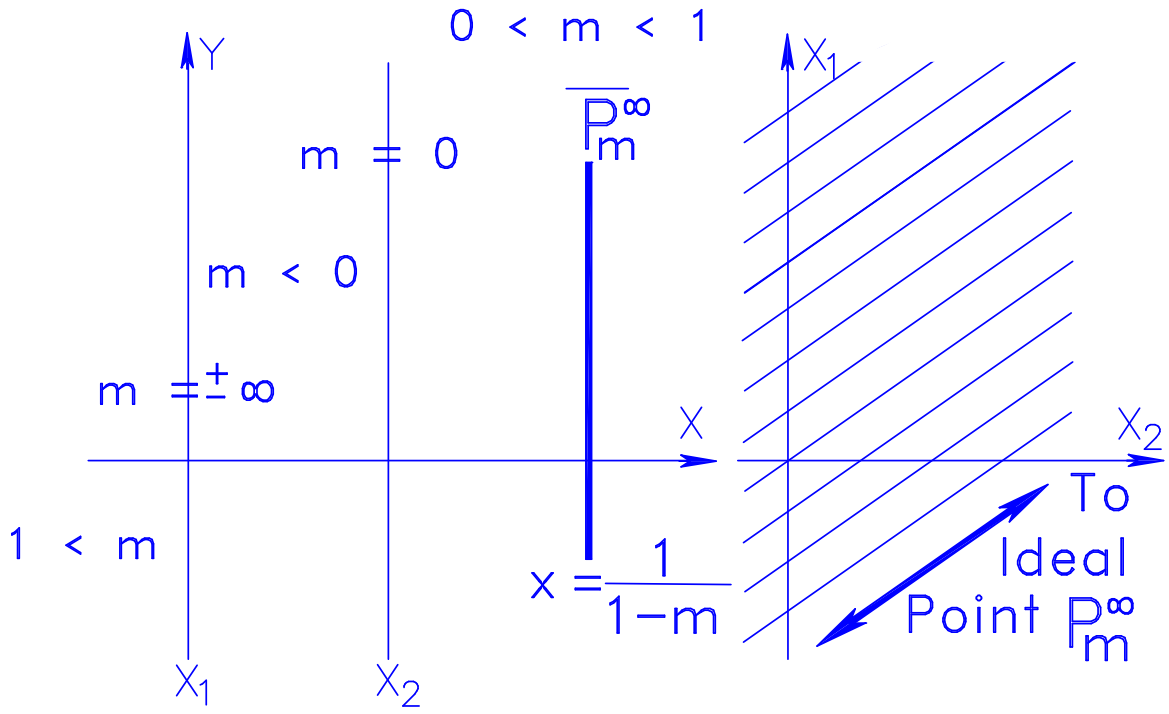

Figure 5: The x-coordinate of $\overline{\ell}$ depends only on the slope of ℓ

farther out but in the specific direction whose slope is b/d. So lines with $m = 1$ are *not mapped into a point* but into a *direction* and still all the information is there! The fact that it is a direction tells us that $m = 1$ while the slope of the direction gives us the value of b. What is going on? Well, dualities properly reside in the *Projective* and not the Euclidean Plane. The "directions" are in fact points (called "ideal") of the *Projective Plane* – which will be described shortly. One does not need expertise in Projective Geometry to do $\|$-coords but awareness is advisable to avoid blunders.

In Fig. 5 we see an important property of the duality, namely the horizontal location of $\bar{l}$ reveals the slope of l. So parallel lines, having the same slope, are represented by points on the same vertical line (see Fig. 7) and that enables us to recognize (i.e. "eyeball") parallelism in $\|$-coords. Further, lines with slope m "meet" at the ideal point denoted by P_m^∞ whose image $\overline{P_m^\infty}$ is the vertical line at $x = 1/(1-m)$.

"Let no one ignorant of Geometry enter" .... At entrance to Plato's Academy

A MODEL OF THE PROJECTIVE PLANE

The Projective Plane can be thought of as the Euclidean Plane with a points at infinity assigned *in every direction*. These are the "ideal" points. The ideal point in the direction with slope m is denoted by P_m^∞. It's image, $\overline{P_m^\infty}$, is the vertical line at $x = 1/(1-m)$ and which represents all parallel lines with slope m. With the stereographic projection shown in Fig. 6 to every point of the Euclidean Plane (i.e. a "regular" point) corresponds a unique point on the hemisphere. Imagining the limiting process as a point goes farther away from the origin (point of tangency with the plane) in a constant direction having slope m, yields that an *ideal point* is represented by the diameter, on the top disk with direction having slope m. Further, as shown in Fig 7, Lines map into great semi-circles. Semi-circles representing parallel lines share the *same diameter* (*i.e. "meet at the ideal point corresponding*

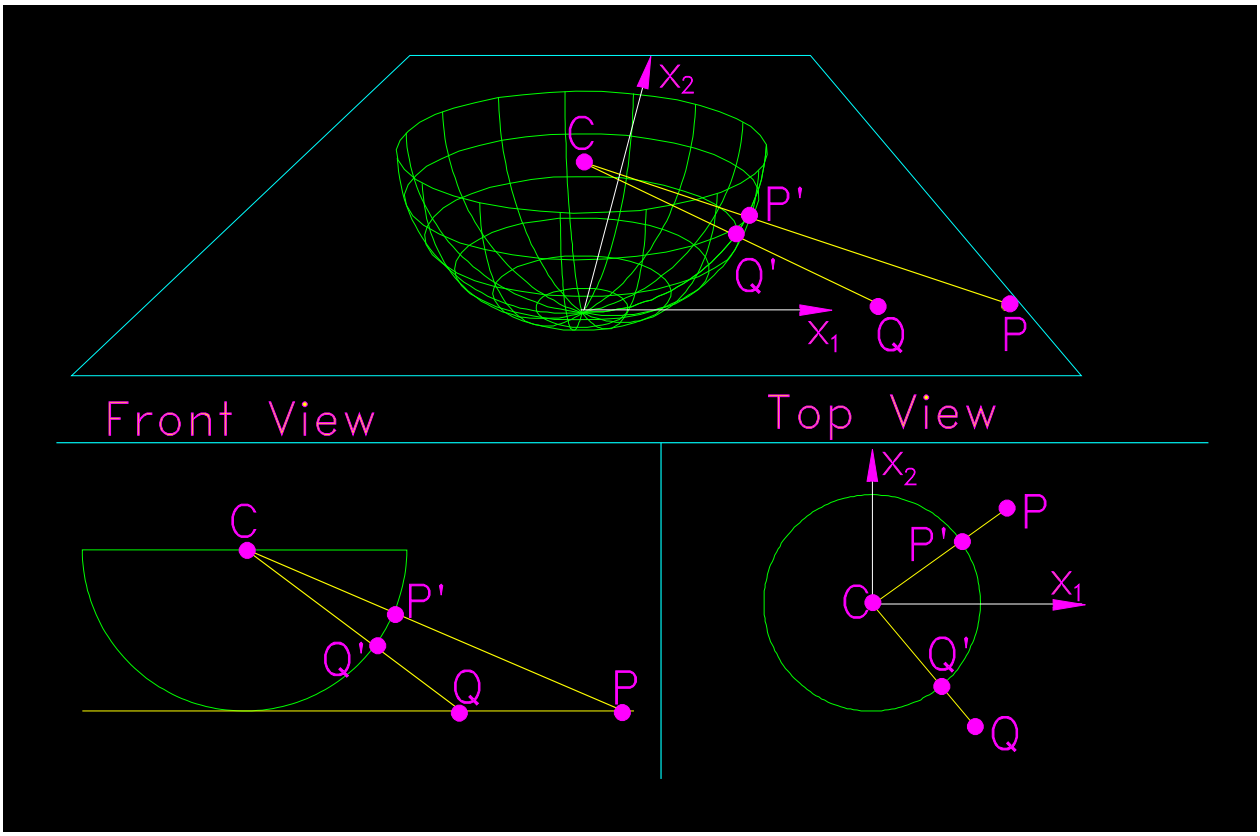

Figure 6: Model of the Projective Plane

to their direction“).

For a change of pace here we will interactively study a financial dataset. We will see some occurrences and the significance of the duality in real data. Also we will discuss the permutations of the axis in the display and discover some surprising evidence about the gold market.

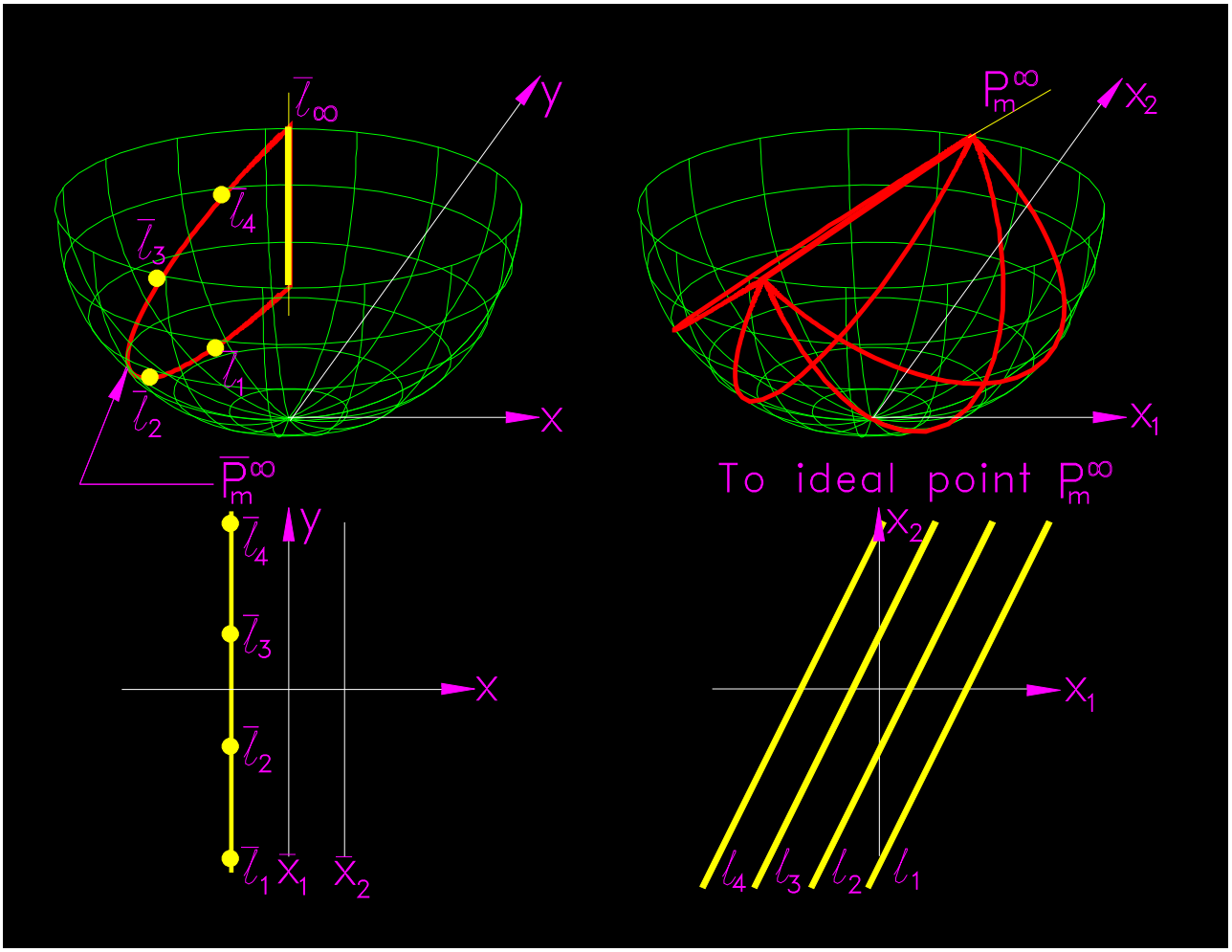

Figure 7: Parallel Lines are represented by points on a vertical line

DETECTING ORTHOGONALITY

With the following two constructions we show that the information on orthogonality is also preserved in $\|$-coords.

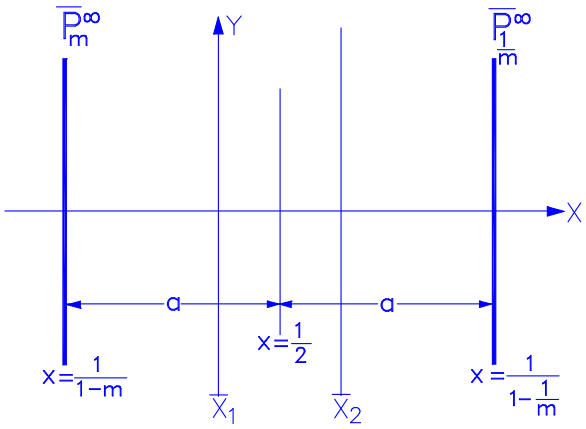

Figure 8: Reflection about $x = 1/2$. Points representing lines with slope m are reflected to points representing lines with slope $1/m$

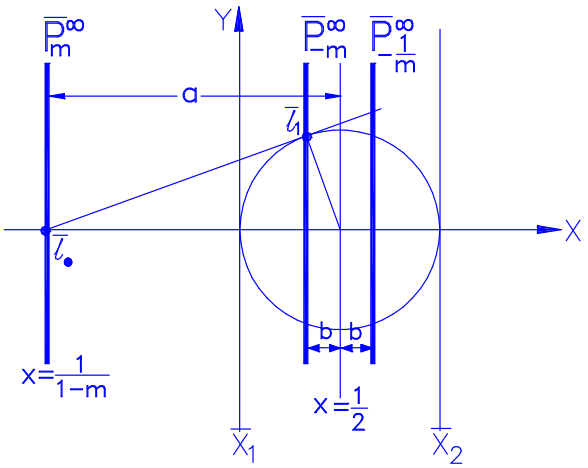

Figure 9: Circle Inversion and Reflection. Points representing lines with slope m are "inverted"to points representing lines with slope $-m$.

Hence the reflection shown in Fig. 8 together with the circle inversion provide the points representing mutually orthogonal families of lines.

THE DUALITY AS A LINEAR TRANSFORMATION

In *Homogeneous Coordinates*

the triple $\ell : [a_1, a_2, a_3]$ **are the** *line coordinates* **of the line**

$a_1 x_1 + a_2 x_2 + a_3 = 0$

which is mapped into the point $\overline{\ell} : (da_2, -a_3, a_1 + a_2)$**.**

Considering the triples for ℓ **and** $\overline{\ell}$ **as column vectors yields the** *correlation* **(not to be confused with the same term used in Statistics – In the language of Projective Geometry this means a linear transformation between line coordinates and point coordinates):**

$\overline{\ell} = A\ell \ , \ \ell = A^{-1}\overline{\ell}$

where

$$A = \begin{bmatrix} 0 & d & 0 \\ 0 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix} \ , A^{-1} = \begin{bmatrix} -1/d & 0 & 1 \\ 1/d & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}$$

and d **is the horizontal distance between the parallel axes**

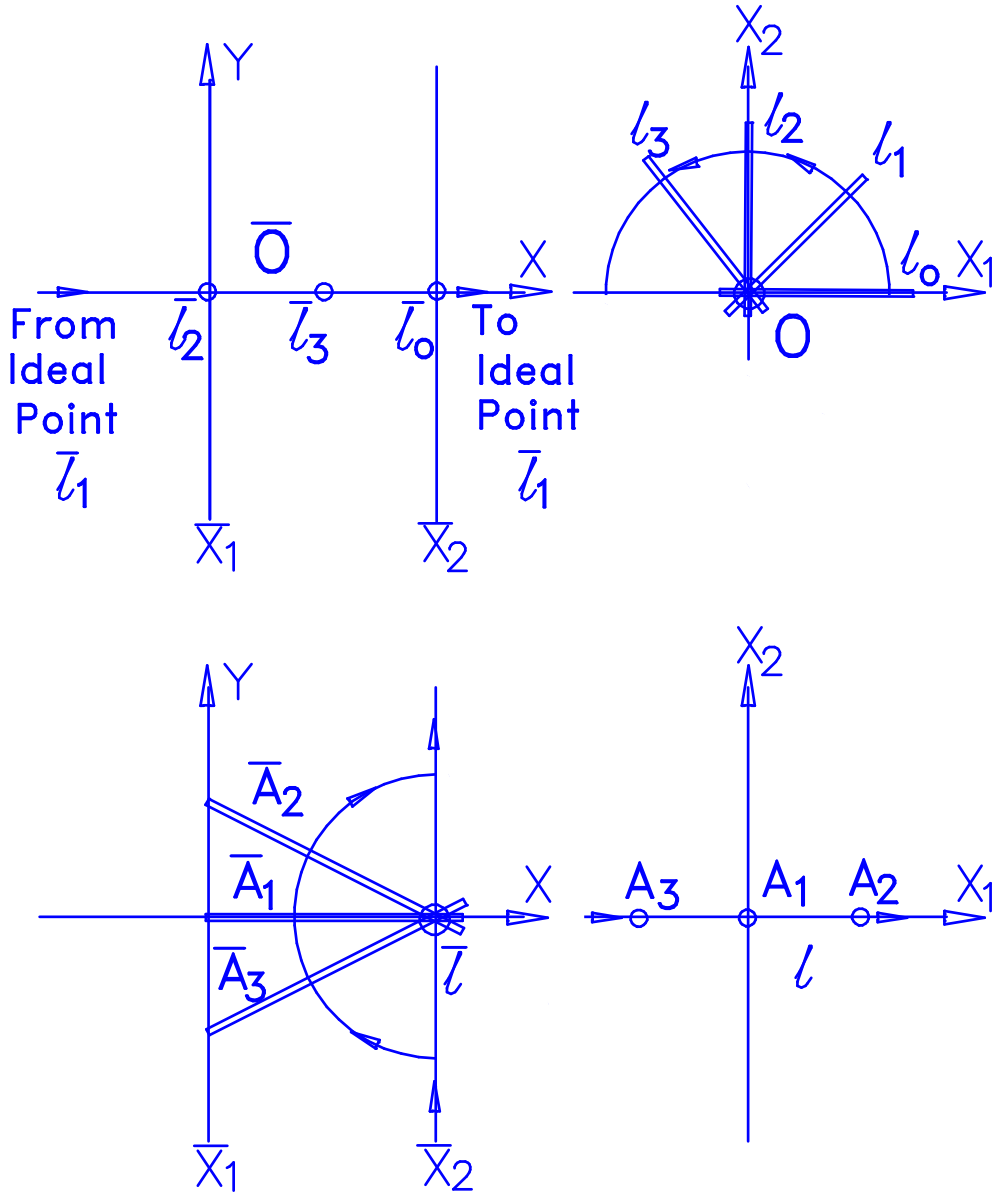

Figure 10: Duality of Transformations

Rotations of a line about a point and translations of a point along a line are dual.

Picture of a square (a), cube in 3-D (b) and Cube in 5-D (c) all having unit side.

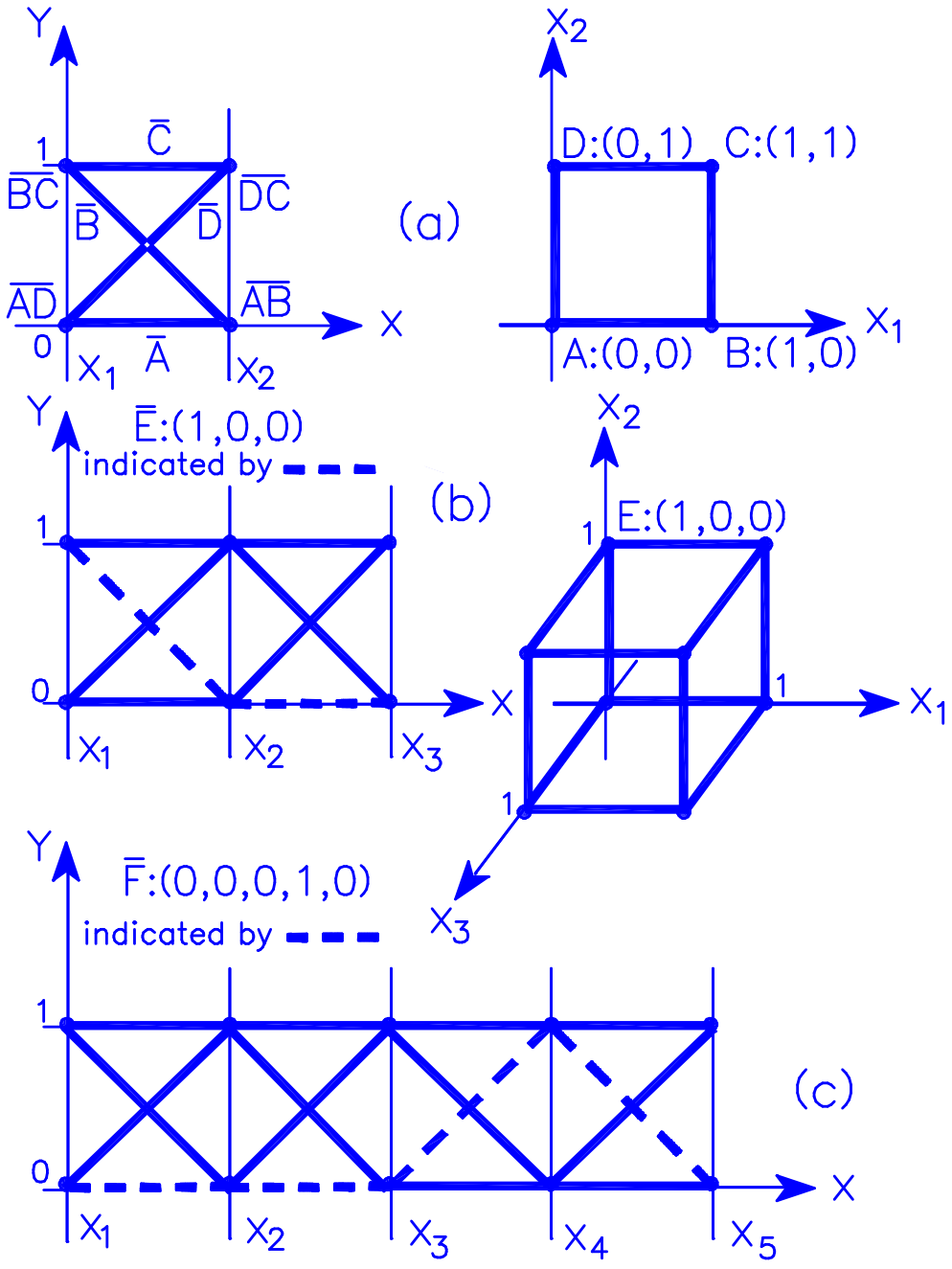

Figure 11: Hypercube Representation in Parallel Coordinates

"Out of nothing I have created a strange new universe"

Bolyai 1823 .... On discovering non-Euclidean Geometry

MULTIDIMENSIONAL LINES

REPRESENTATIONS & CONSTRUCTION ALGORITHMS

♢ **A line ℓ in R^N is represented by $N-1$ points with two indices $i, j \in [1, 2, \ldots, N]$.**

♢ **There are two common ways to describe lines.**

♢ **Either in terms of *ad jacent* variables :**

$$\ell_{1,2} \; : \; x_2 \; = \; m_2 x_1 + b_2$$

$$\ell_{2,3} \; : \; x_3 \; = \; m_3 x_2 + b_3$$

$$\ldots$$

$$\ell_{i-1,i} \; : \; x_i \; = \; m_i x_{i-1} + b_i$$

$$\ldots$$

$$\ell_{N-1,N} \; : \; x_N \; = \; m_N x_{N-1} + b_N$$

♢ **or in terms of a single variable, the** *base variable* **which can be taken as** x_1**, i.e.**

$$\begin{aligned}
\ell_{1,2} \ : \ x_2 &= m_2^1 x_1 + b_2^1 \\
\ell_{1,3} \ : \ x_3 &= m_3^1 x_1 + b_3^1 \\
&\dots \\
\ell_{1,i} \ : \ x_i &= m_i^1 x_1 + b_i^1 \\
&\dots \\
\ell_{1,N} \ : \ x_N &= m_N^1 x_N + b_N^1
\end{aligned}$$

♢ **The** $N-1$ **indexed points (in homogeneous coordinates) are :**

♢ **in the first case**

$$\overline{\ell}_{i-1,i} = ((i-2)(1-m_i) + 1,\ b_i,\ 1-m_i)\ ,$$

♢ **and in the second case**

$$\overline{\ell}_{1,i} = (i-1,\ b_i^1,\ 1-m_i^1)\ .$$

The *indexing* **of the points is an essential part of the representation and it is crucially used in the subsequent algorithms. Though the indexing is often not shown to save display space, it needs to be accessible.**

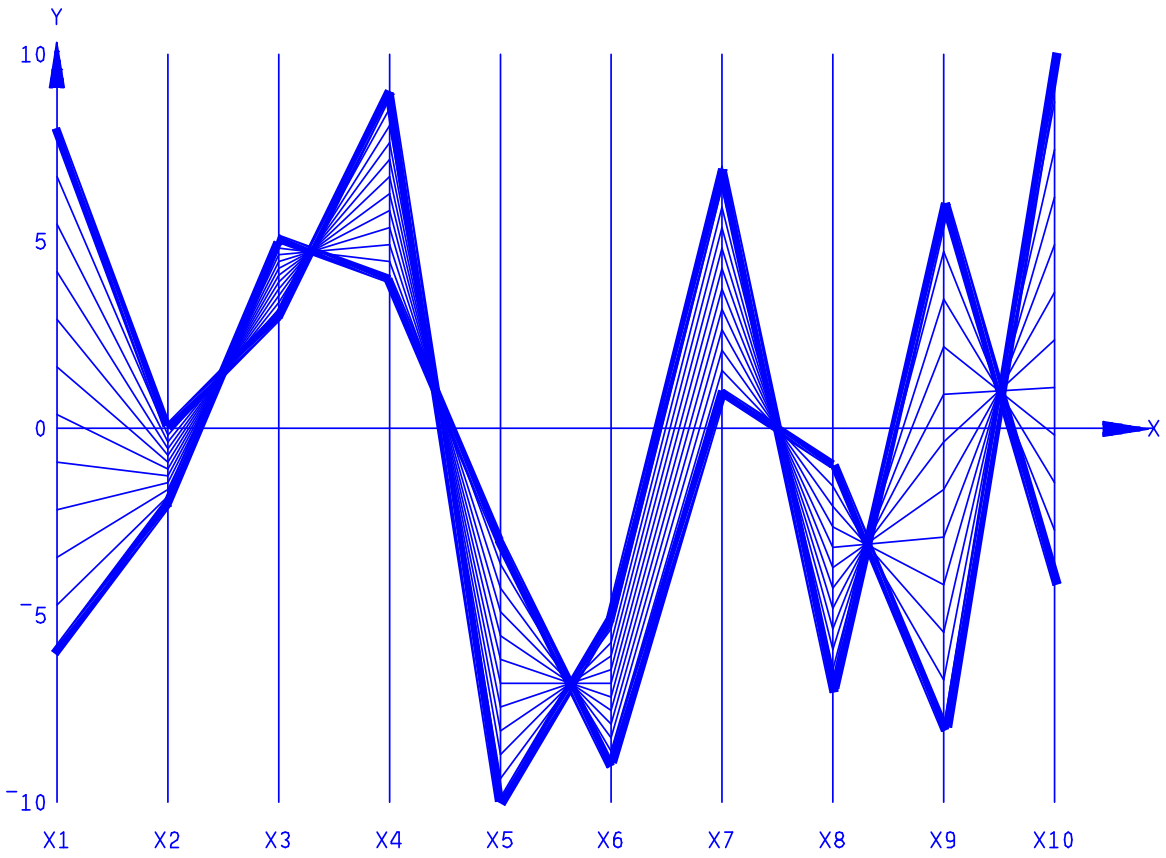

Figure 12: Interval on a line in R^{10}

The polygonal lines through all the points represent points on on the line with the heavier polygonal lines indicating the endpoints. The points shown here representing the line correspond to the adjacent variables parametrization. Here the indexing, of the points representing the line, can be found from the intersecting segments of the polygonal line. For example, if $\bar{\ell}_{1,2}$ was shown it would lie to the right of the $X2$ axis, $\bar{\ell}_{7,8}$ lies between the $X7$ and $X8$ axes etc.

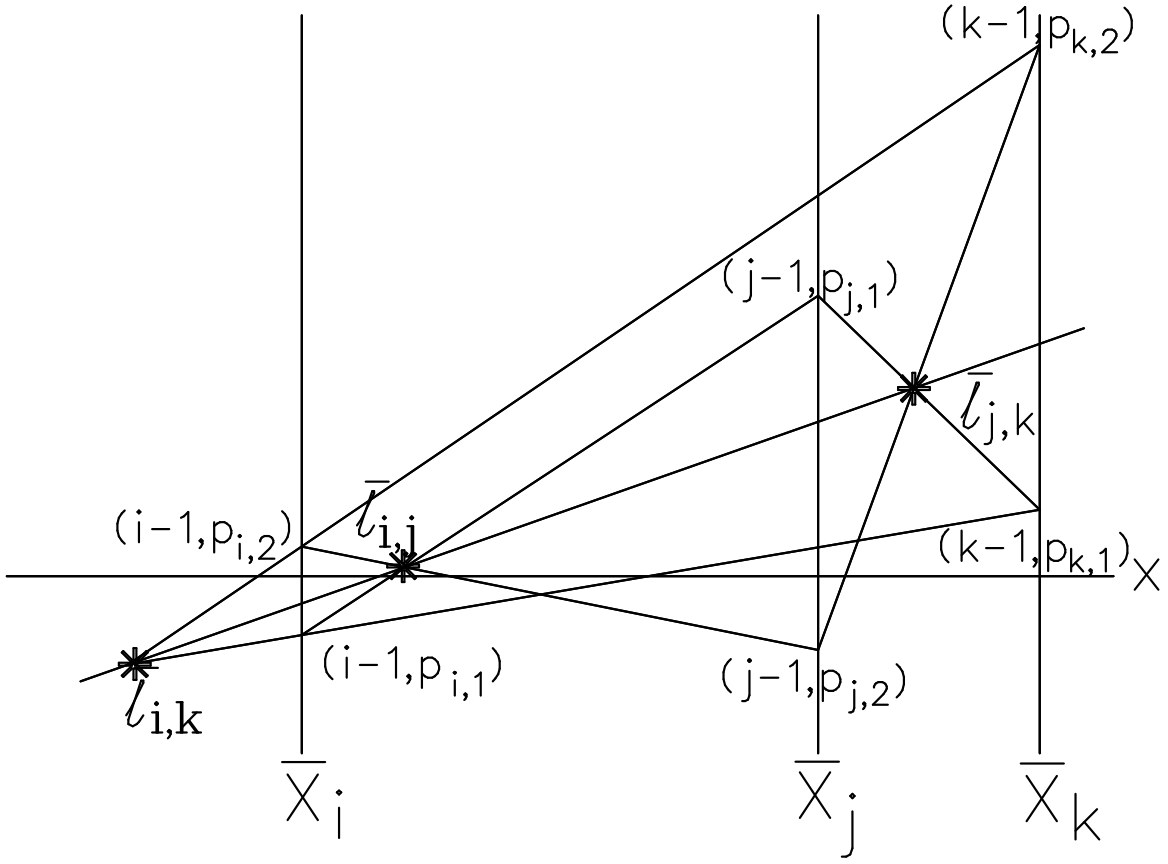

Figure 13: Collinearity of the points $\bar{\ell}_{i,j}$, $\bar{\ell}_{j,k}$, $\bar{\ell}_{i,k}$.

The three point collinearity property plays a fundamental role in the representation algorithms for higher dimensional objects. It is found by an application of Desargues theorem of Projective Geometry . The two triangles shown are in perspective with respect to the ideal point in the vertical direction.

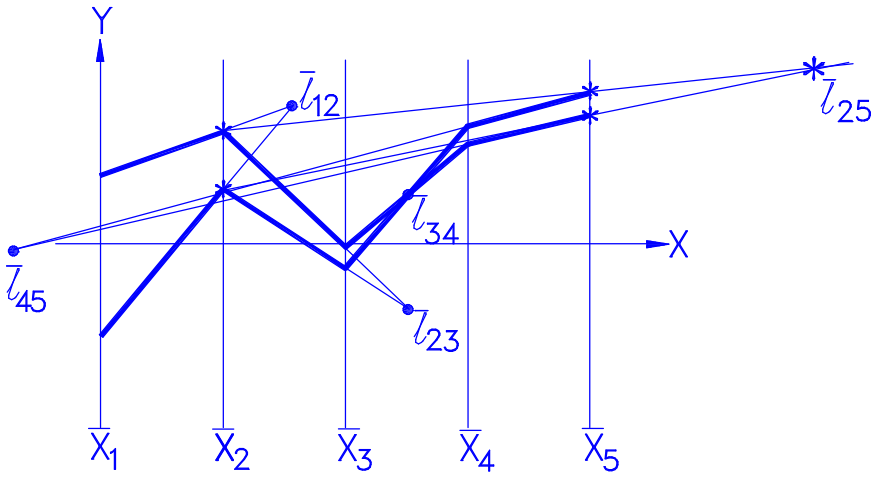

Figure 14: The point $\bar{\ell}_{2,5}$ found by construction

For a line ℓ in R^N the linear relation between any pair of variables can be found *geometricaly* from the $N-1$ points representing the line. Here $\bar{\ell}_{2,5}$ is constructed as the intersection of the segments joining the coordinates of two points (on the line) on the $\bar{X}_2$ and $\bar{X}_5$ axes.

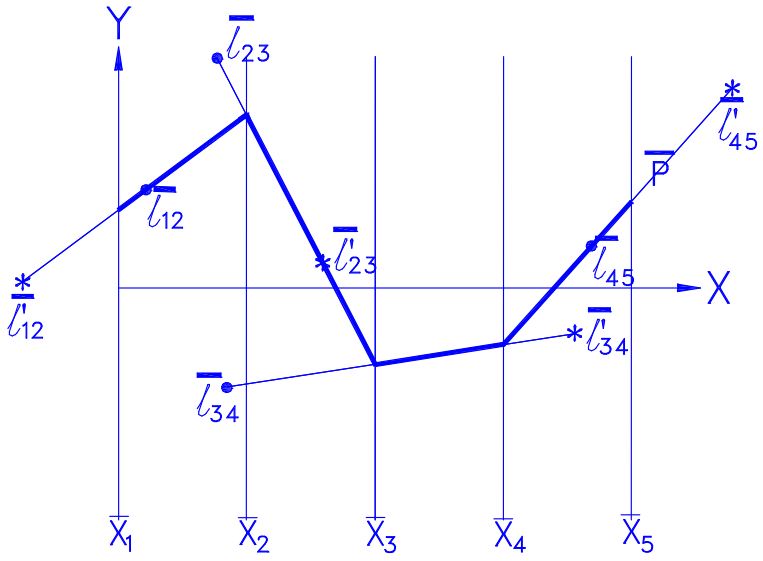

Figure 15: Rotation of a line about one of it's points

"It was recently discovered that the rings of Saturn are made of lost airline luggage ... that's why more of them are discovered every week!"

DISPLAYING AIR TRAFFIC INFORMATION

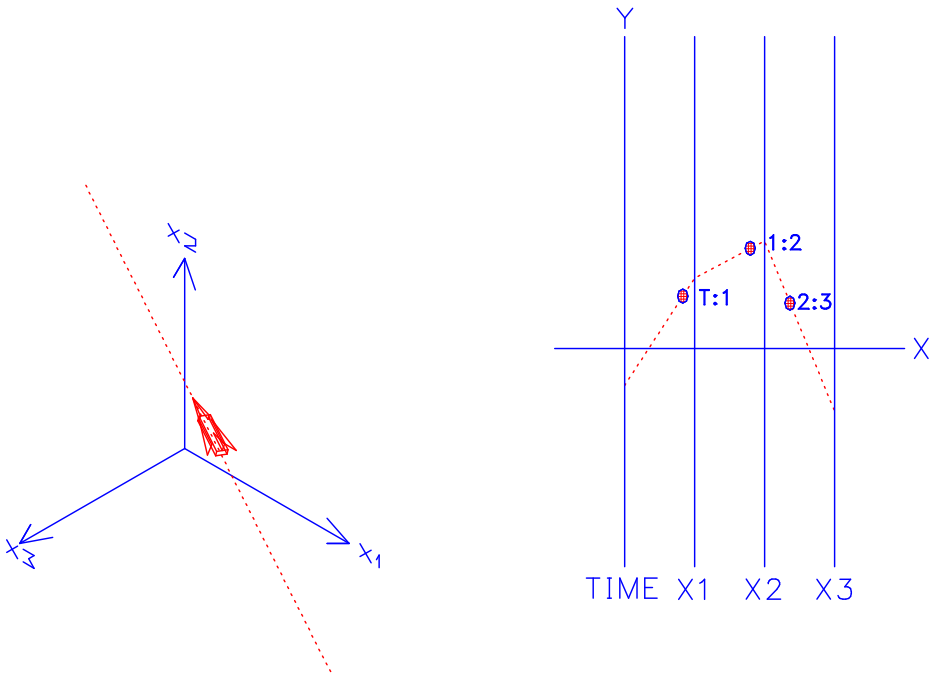

Figure 16: Path(left) and trajectory(Right) of an aircraft

The 3-D picture shows the path and position while the polygonal line in parallel coordinates shows the position at a given time. The point $T:1$ represents the linear relation between time T and the X_1-coordinate while $1:2$ and $2:3$ represent the path, i.e. the pairwise linear relation between x_1 , x_2 , x_3 .

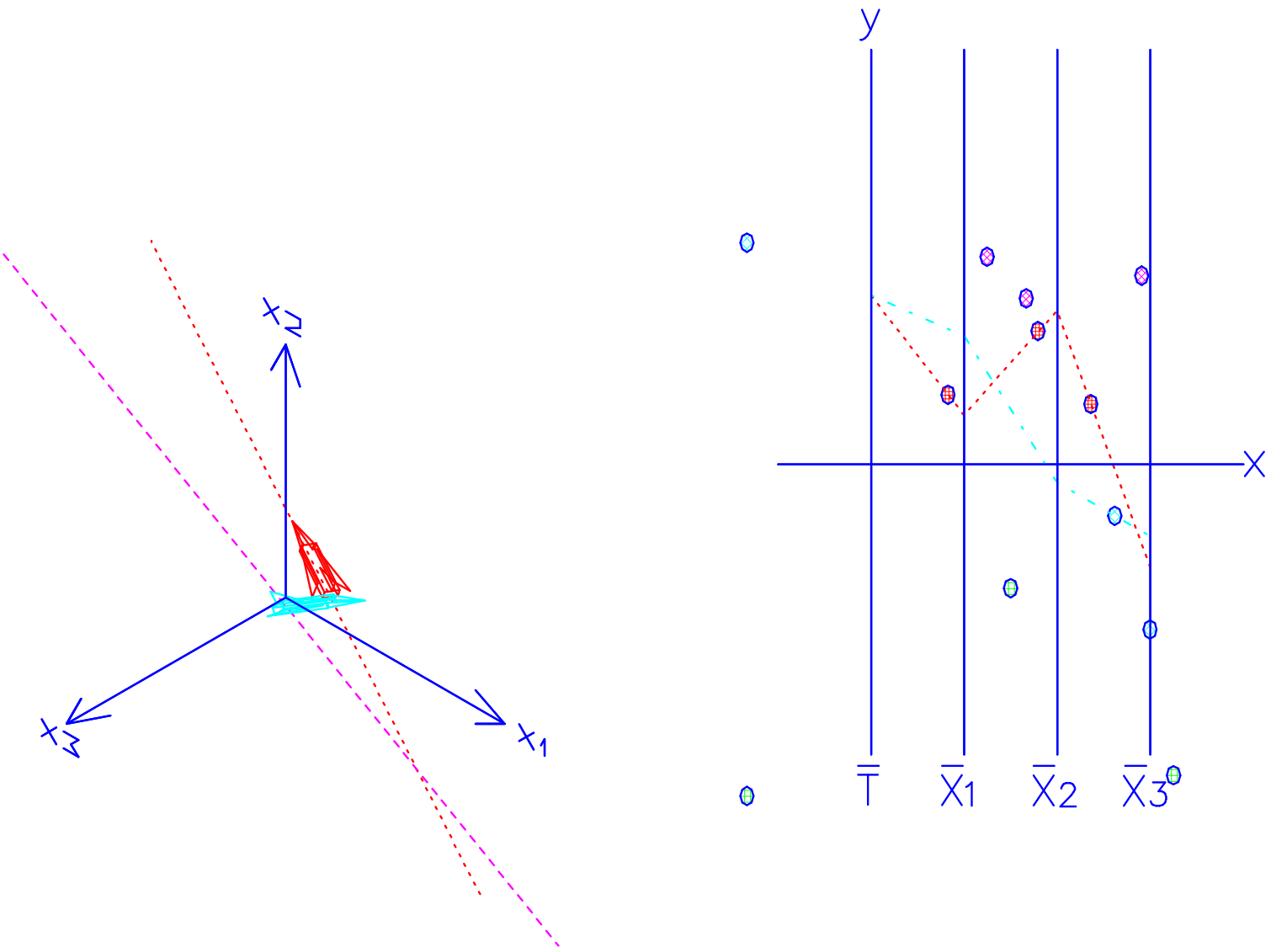

Figure 17: Closest approach of two aircraft

The time at which this occurs *and* their corresponding positions. On the four parallel axes a polygonal line shows the time, value on the T-axis, when the two positions in (x_1 , x_2 , x_3) is attained.

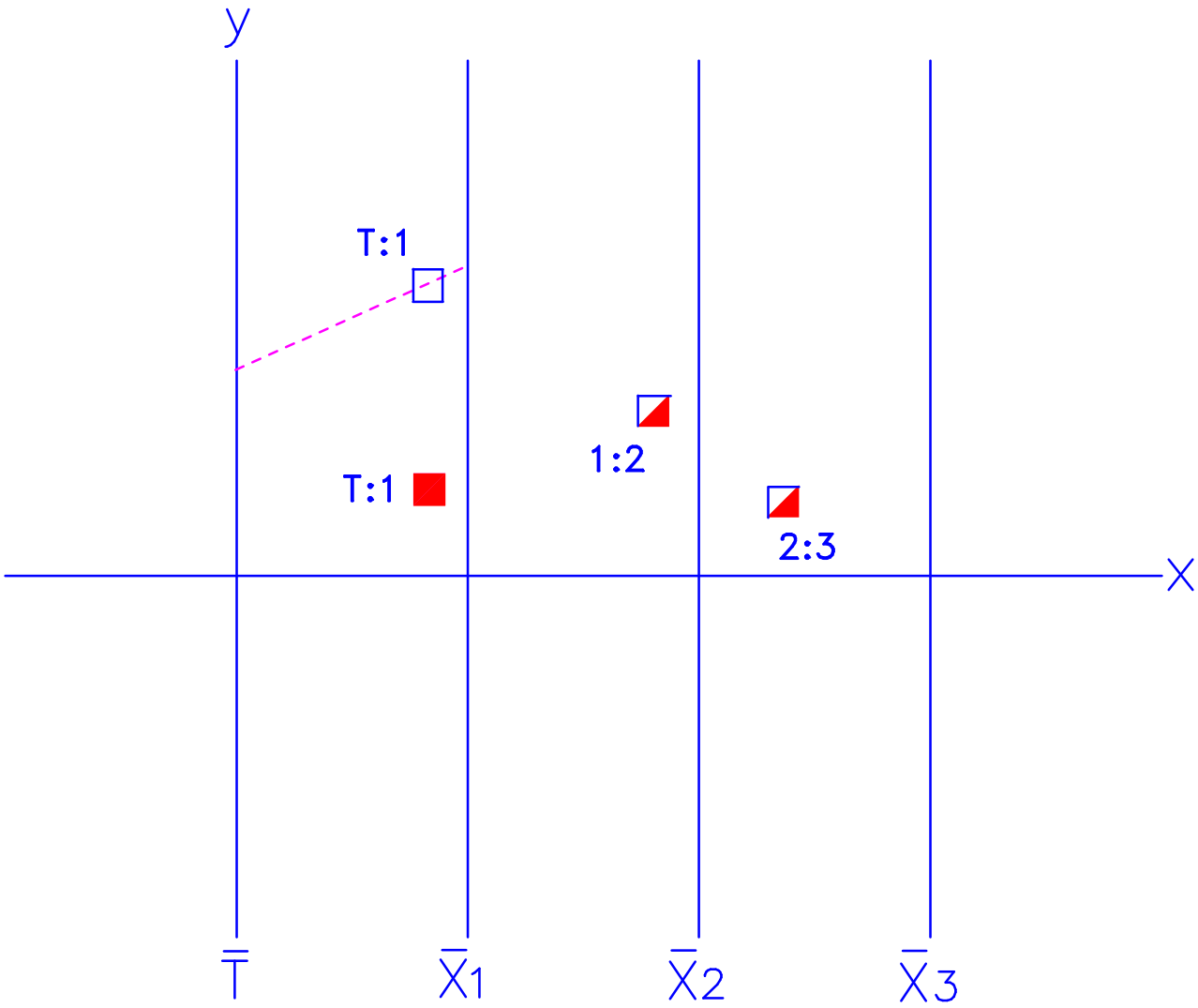

Figure 18: Two aircraft flying the same path with the same velocity

Note that the $1:2\ ,\ 2:3$ **points indicated by boxes (these are the** $\bar{l}$**'s) are shared indicating that the paths in 3-D are the same. When that occurs, the leftmost** $T:1$ **corresponds to the greater speed. Here the airplanes have the same velocity since the two** $T:1$ **points have the same horizontal position.**

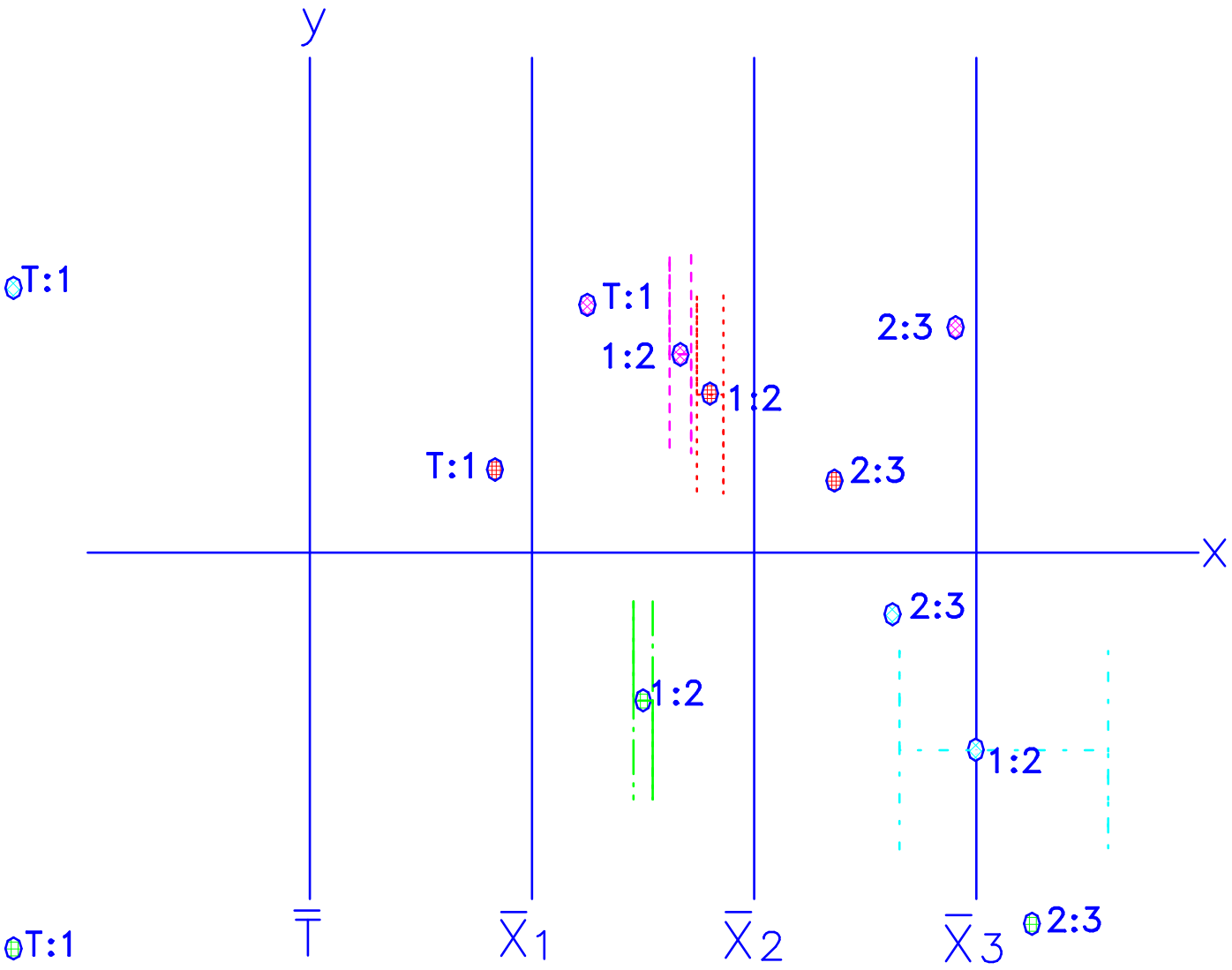

Figure 19: Angular deviations for assigned trajectories

A deviation of $\pm\theta$ degrees transforms into a lateral deviation centered about the appropriate point. Here a deviation of ± 5 degrees in ground heading is shown when x_3 is the altitude scale.

DISTANCE & PROXIMITY PROPERTIES

INTERSECTIONS

∇ **The pair of lines $\ell,\ \ell'$ given by**

$$\ell, \ell_{1i} \ :\ x_i \ = \ m_i x_1 \ + \ b_i$$

$$\ell', \ell'_{1i} \ :\ x_i \ = \ m'_i x_1 \ + \ b'_i,$$

$$i \ = \ 2, \ \ldots \ , N$$

∇ **and represented in xy-coordinates by the points**

$$\overline{\ell}_{1i} \ :\ x \ = \ \frac{i-1}{1-m_i} \ , \ y \ = \ \frac{b_i}{1-m_i}$$

$$\overline{\ell'}_{1i} \ :\ x \ = \ \frac{i-1}{1-m'_i} \ , \ y \ = \ \frac{b'_i}{1-m'_i}$$

∇ **intersect with $\ell \cap \ell'$ = P $\Leftrightarrow$**

$$\alpha_i \ = \ \frac{b'_i - b_i}{m'_i - m_i} \ = \ p_1 \ , \ \forall i \ = \ 2, \ \ldots \ , N$$

∇ **where $x_1(P) \ = \ p_1$.**

∇ **Analogous criteria exist for different parametrizations.**

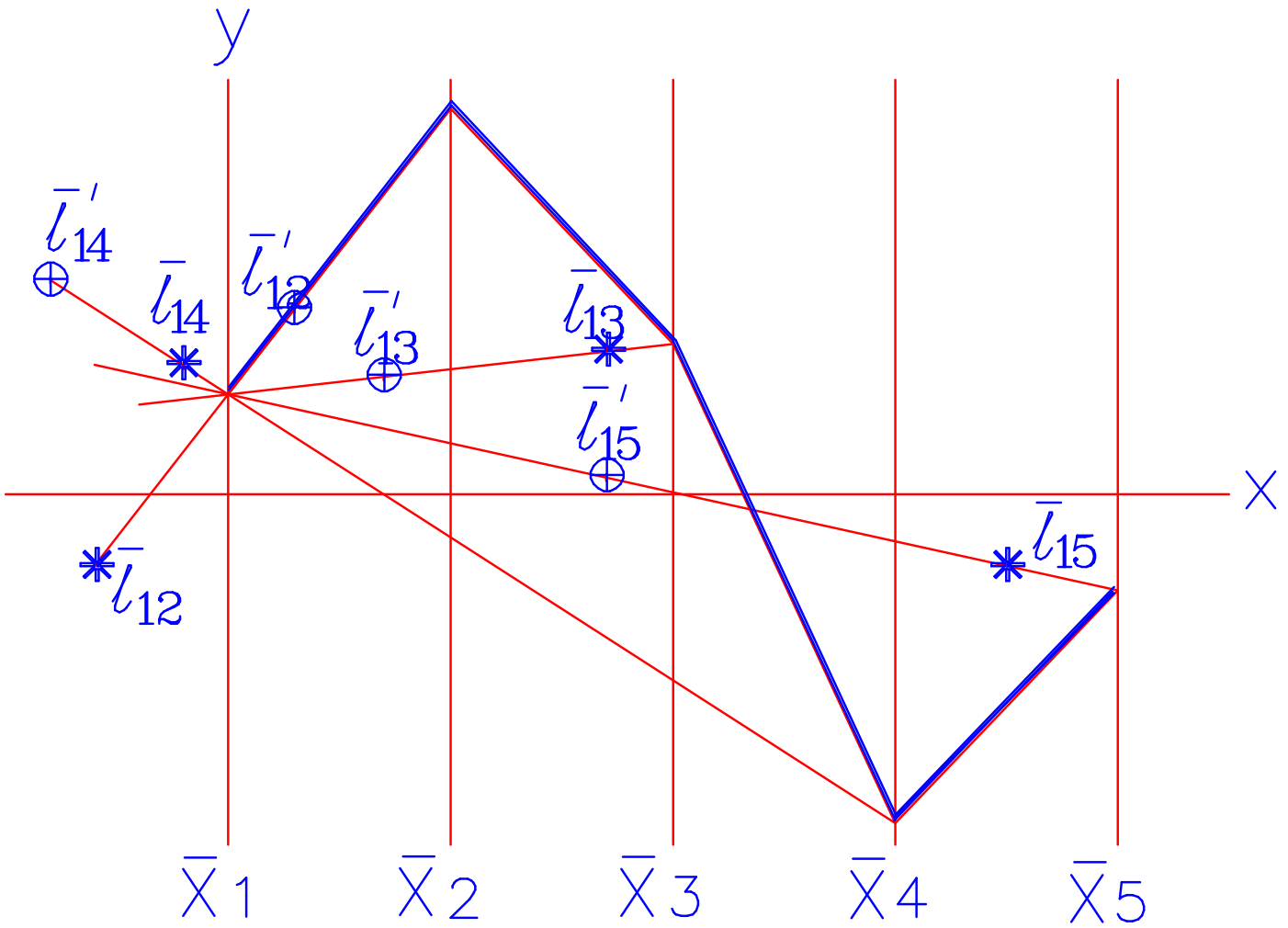

Figure 20: Two lines intersecting in R^5 - first example

Common parameter is x_1. One line, ℓ is represented by the points $\overline{\ell}_{1i}$ and the other, ℓ' by $\overline{\ell}'_{1i}$ for $i = 2, 3, 4, 5$. The two lines intersect $\Leftrightarrow$ the lines $\overline{P}^{1i}$, joining $\overline{\ell}_{1i}$ and $\overline{\ell}'_{1i}$, intersect at the same point of the x_1-axis. The polygonal line represents the point of intersection.

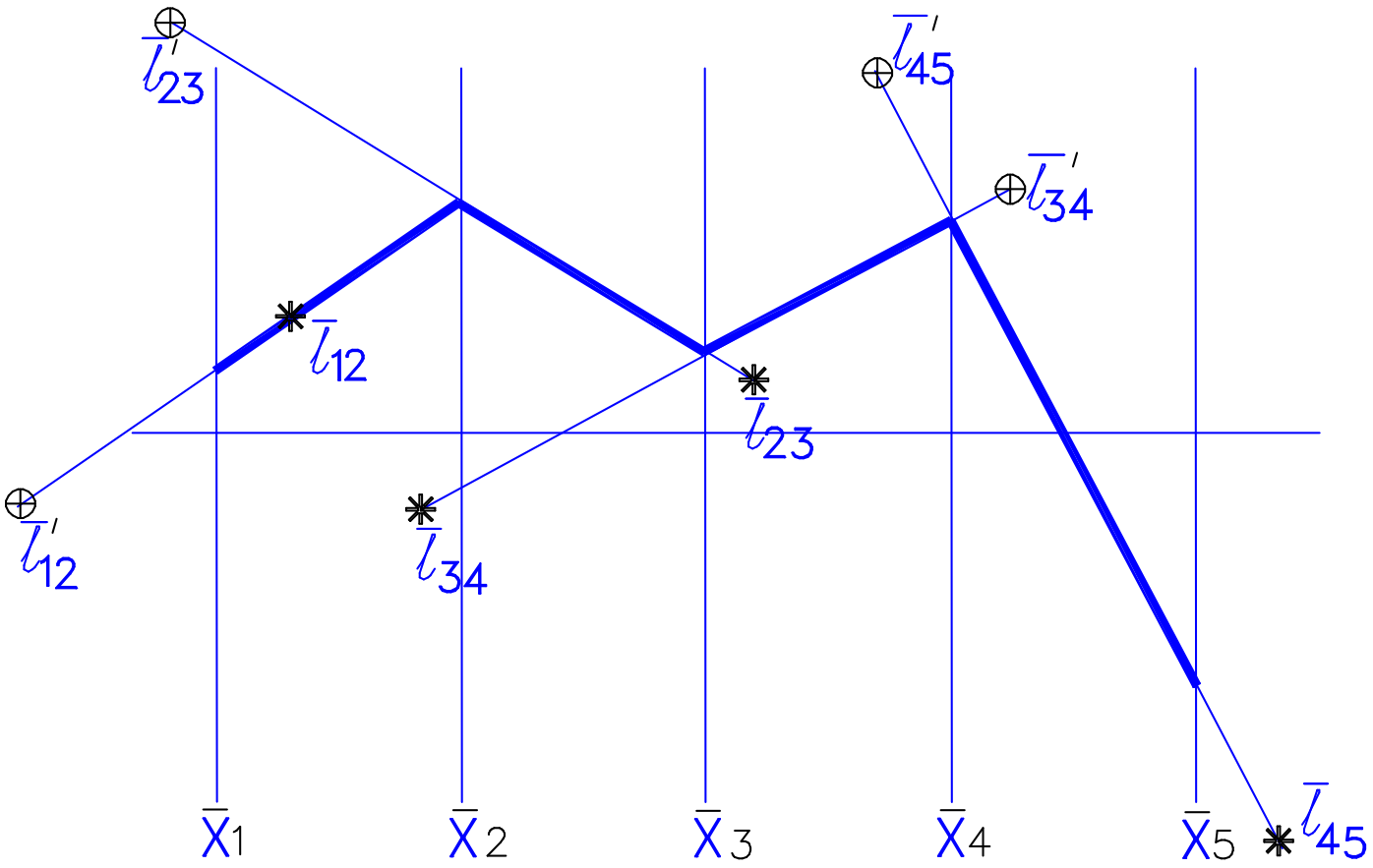

Figure 21: Two lines intersecting in R^5 - second example

Here the representation is based on consecutive adjacent pairs of coordinates. The two lines,$\ell,\ \ell'$ are represented by the points $\overline{\ell}_{i\,,\,i+1}$ and $\overline{\ell}'_{i\,,\,i+1}$ $i\ =\ 1,\ 2,\ 3,\ 4$. The two lines intersect $\Leftrightarrow$ the line $\overline{P}^{i,i+1}$, joining $\overline{\ell}_{i\,,\,i+1}$ and $\overline{\ell}'_{i\,,\,i+1}$, intersects the x_i-axis at the same point as the line $\overline{P}^{i+1,i+2}$, joining joining $\overline{\ell}_{i+1\,,\,i+2}$ and $\overline{\ell}'_{i+1\,,\,i+2}$, for all i. The polygonal line shown represents the point of intersection.

"Law of Attraction of Unfortunate Events"

"Unfortunate events tend to attract others of their kind."....................

MINIMUM DISTANCE BETWEEN TWO LINES

△ **For the pair of lines ℓ , ℓ' previously described in terms of the base variable x_1,**

△ **the L_1 distance between two points one on each of the lines is given by :**

$$L_1(x_1) \;=\; \sum_{i=2}^{N} \mid x_i - x_i' \mid = \; \sum_{i=2}^{N} \mid \Delta m_i \mid \mid x_1 - \alpha_i \mid$$

△ **The minimum of $L_1(x_1)$ occurs at an $x_1 \;=\; \alpha_i$.**

△ **The value of x_1 at which the minimum Euclidean distance L_2 occurs is**

$$\alpha^* \;=\; \frac{\sum \alpha_i \Delta m_i^2}{\sum \Delta m_i^2}$$

where the summation is only over those values of i where $\Delta m_i \;\neq\; 0$.

△ **It turns out that the minimum L_1 occurs very close or at α^*.**

For comparison the minimum L_2 distance occurs at $x_1 \;=\; \alpha^*$. The $\mid \Delta m_i \mid$ are added on the bar chart (to the right of the x_6 axis) in the order $6,\ 2,\ 4,\ 3,\ 6$ obtained from the order of increasing α (as shown on the x_1-axis). The I where the mid-point value of the $\sum \mid \Delta m_i \mid$ occurs provides the correct $x_1 \;=\; \alpha_I$. Here $\mid \Delta_4 \mid$ dominates the sum yielding $I \;=\; 4$.

All joint intercepts are equal.

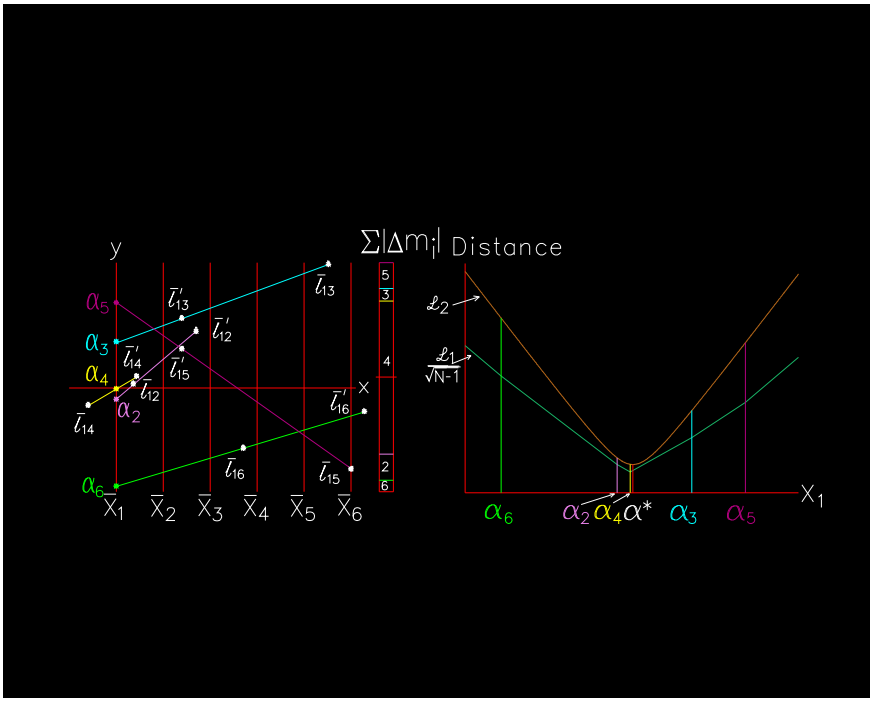

Figure 22: Finding $x_1 = \alpha_I$ minimizing the L_1 distance between two lines

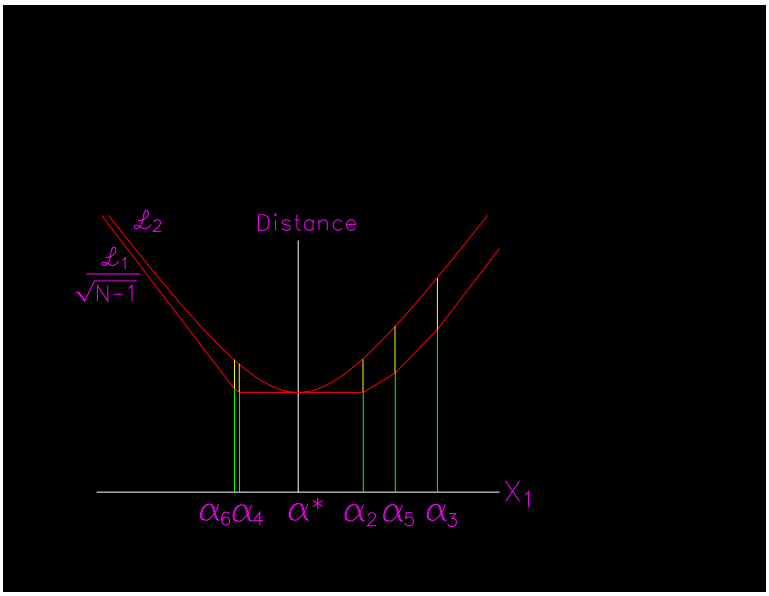

Figure 23: Here the L_1 and L_2 minima coincide

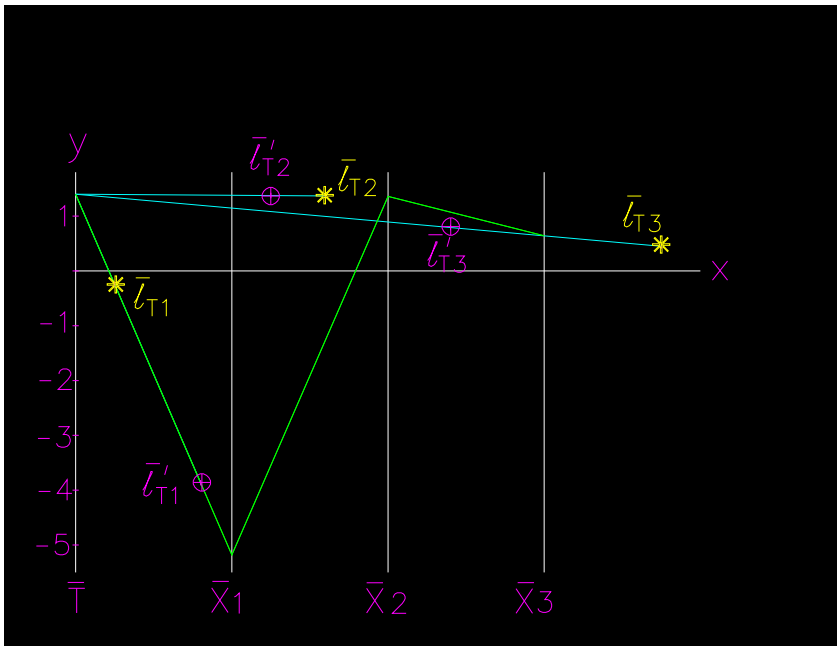

Figure 24: Intersecting lines in 4-D

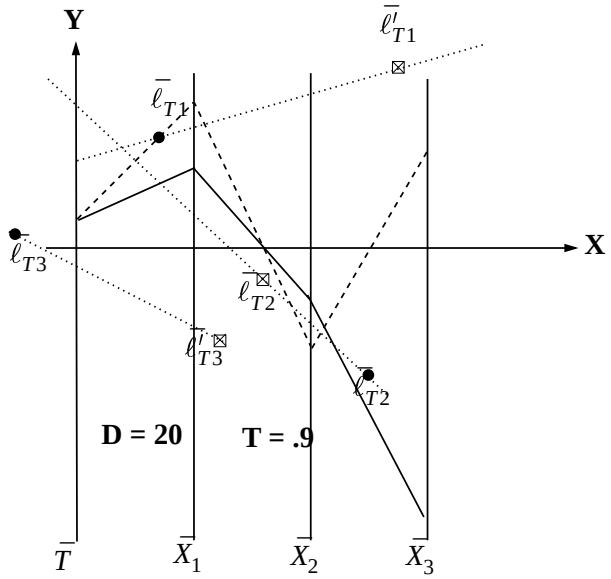

Figure 25: Non-intersection between two lines in 4-D. Here the minimum distance is 20 and occurs at time = .9. Note the maximum gap on the $\bar{T}$-axis formed by the lines joining the $\bar{\ell}$'s with the same subscript. The polygonal lines representing the points where the minimum distance occurs are shown.

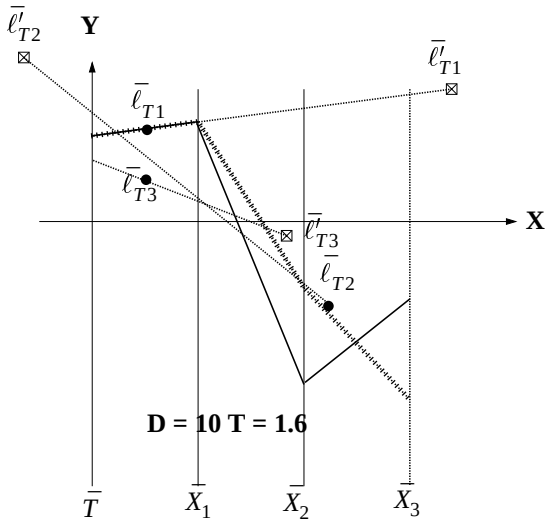

Figure 26: Non-intersection between two lines in 4-D. Here the minimum distance is 10 and occurs at time = 1.6. Note the the diminishing maximum gap on the $\bar{T}$-axis formed by the lines joining the $\bar{\ell}$'s with the same subscript and compare with Fig. 25. The polygonal lines representing the points where the minimum distance occurs are shown.

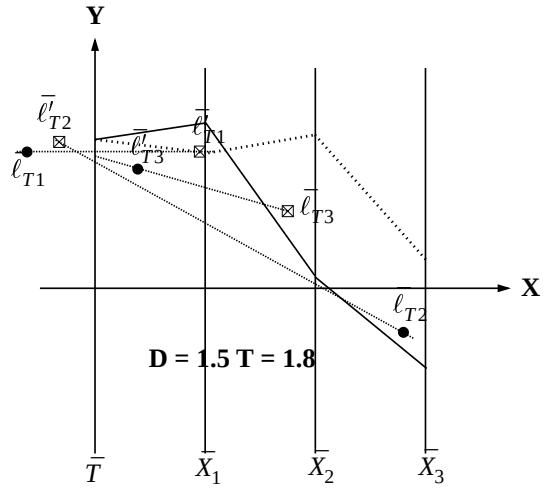

Figure 27: Near intersection between two lines in 4-D. Here the minimum distance is 1.5 and occurs at time = 1.8. Note the the diminished maximum gap on the $\bar{T}$-axis formed by the lines joining the $\bar{\ell}$'s with the same subscript. The polygonal lines representing the points where the minimum distance occurs are shown.

"Law of Inopportune Timing"

"If a fortunate event occurs at all it tends to happen a bit too soon or a bit too late"

CONFLICT DETECTION & RESOLUTION FOR AIR TRAFFIC CONTROL

THE BASIC ALGORITHM

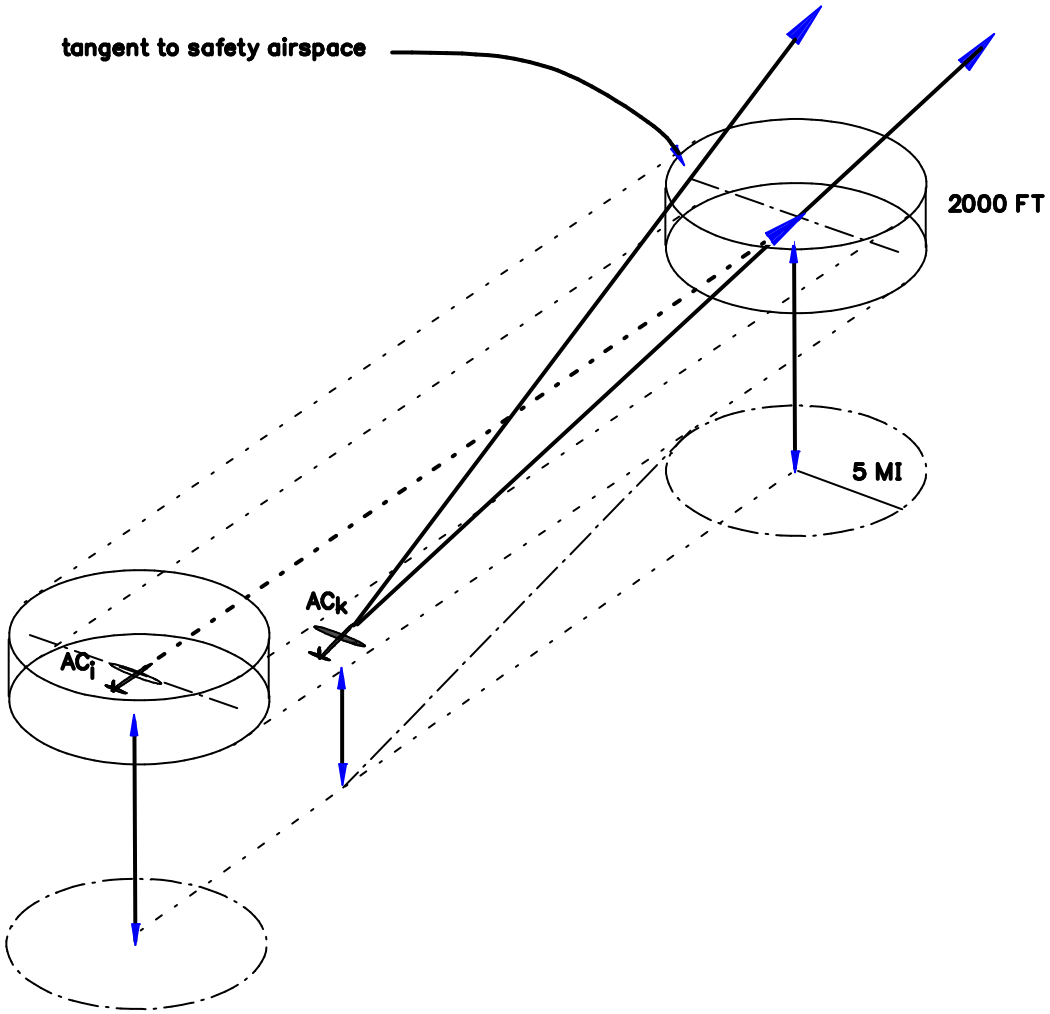

Figure 28: Protected airspace in 3-D

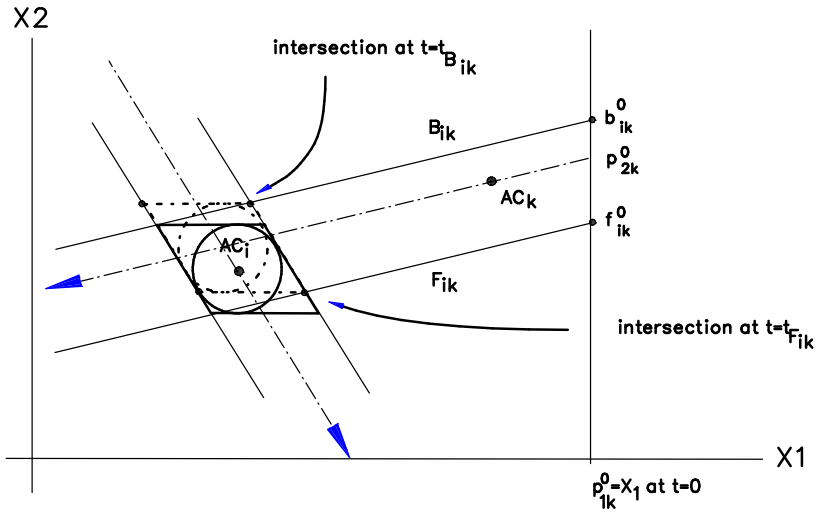

Figure 29: Determining the front and back scrapes

At $t = 0$ all particles on the vertical line $x_1 = p^0_{1k}$ are poised to move with the same velocity as AC_k. Particles which just scrape the circle from the front (i.e. starting at $x_2 = f^0_{ik}$) and back (i.e. starting at $x_2 = b^0_{ik}$ define the *limiting trajectories*. Here there is a conflict between AC_k and AC_i since $f^0_{ik} < p^0_{2k} < b^0_{ik}$.

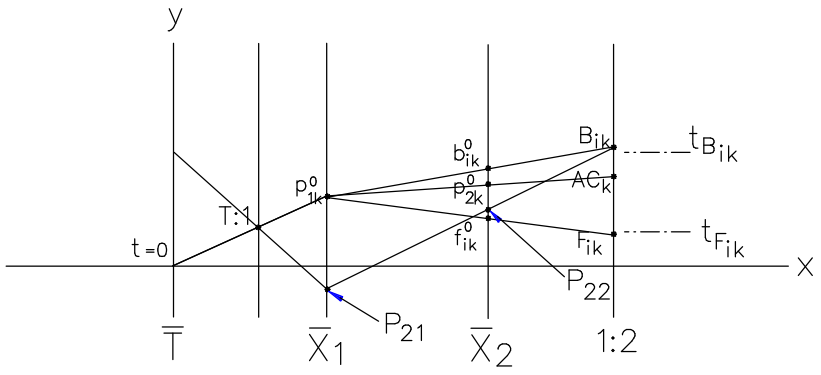

Figure 30: The limiting trajectories (scrapes) information in parallel coordinates

The ordinates of the points P_{21} and P_{22} are the coordinates of the intersection of B_{ik} with the conflict parallelogram.

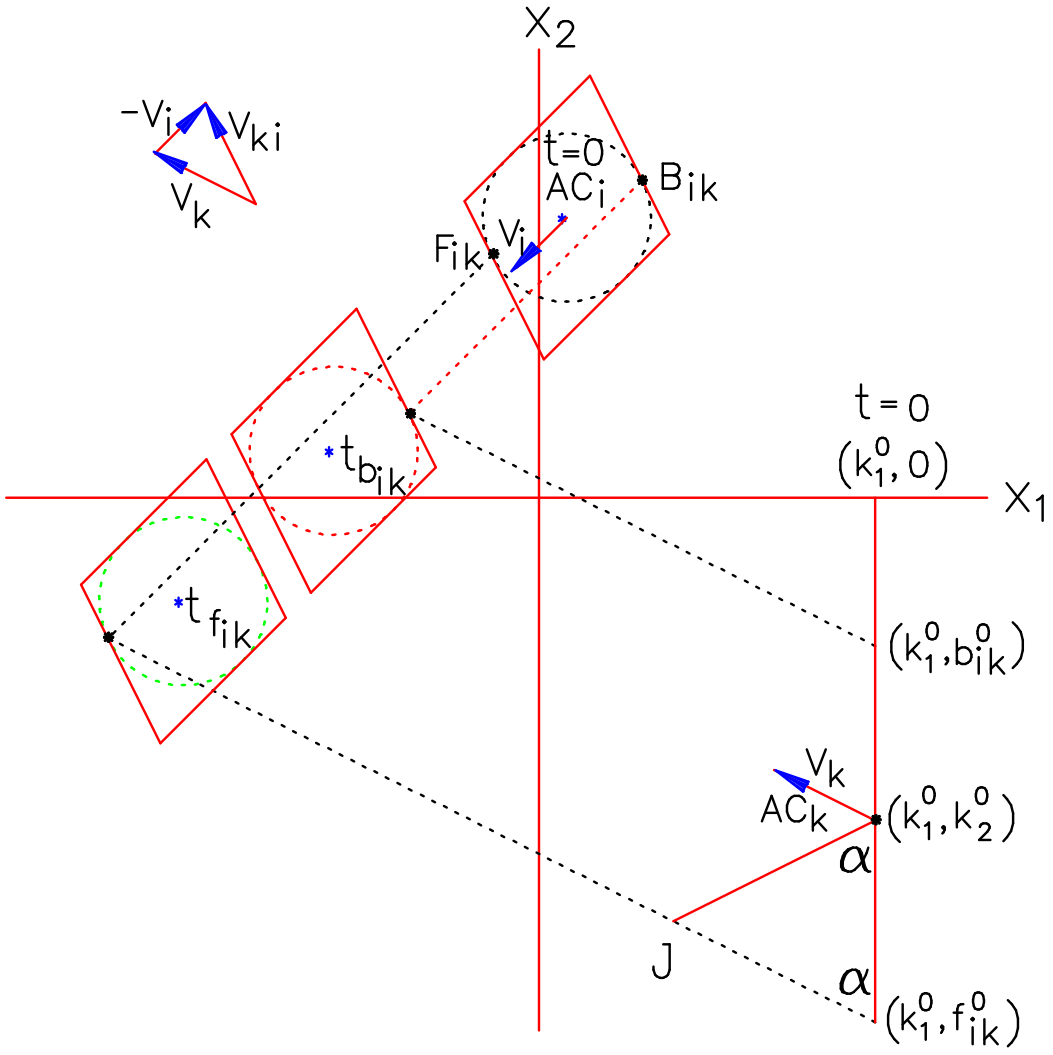

Figure 31: Relation Between Maneuver-Speed and Turn-Angle

Maneuver *with no speed change* **can be done with turn angle** α**. Turn angles greater or less than** α **require a slower or faster speed respectively than** $|V_k|$**.**

RESOLUTION OF A CONFLICT SCENARIO

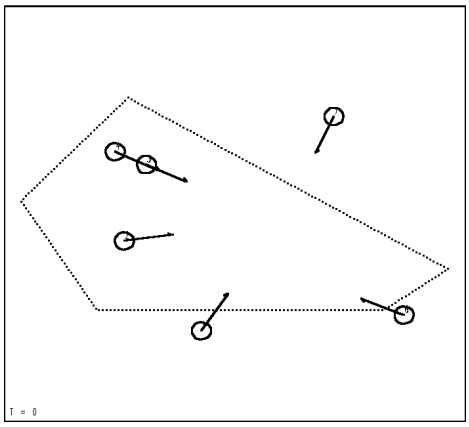

Figure 32: Six aircraft from scenario flying at the same altitude

Initial positions (T = 0 sec.) and circles centered at each aircraft with radius 2.5 nm (5 nm separation standard) are shown to scale with arrows representing velocity vectors.

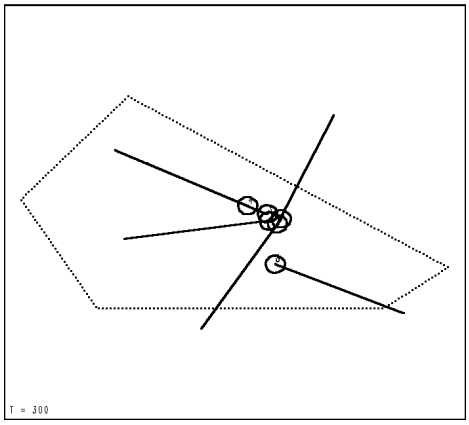

Figure 33: Conflicts among the six aircraft

A conflict occurs when the separation between any two aircraft is less than 5 nm (i.e. two circles intersect). Several conflicts occur within the first 5 minutes (time elapsed in seconds is indicated in the lower left hand corner).

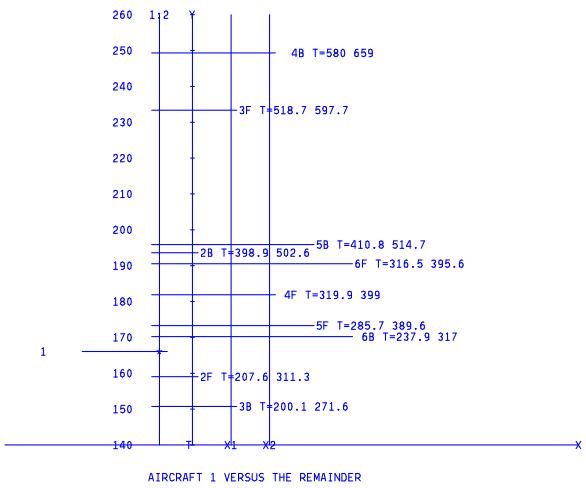

Figure 34: Conflict intervals (CI)

Using the data of conflict scenario the CI_{1k} , $k = 2,3,4,5,6$ (i.e. of aircraft 2,3,4,5,6 versus 1) are plotted. Vertical scale units of distance are representing specific paths parallel to those of aircraft 1. Times shown indicate entry and exit from corresponding conflict parallelogram.

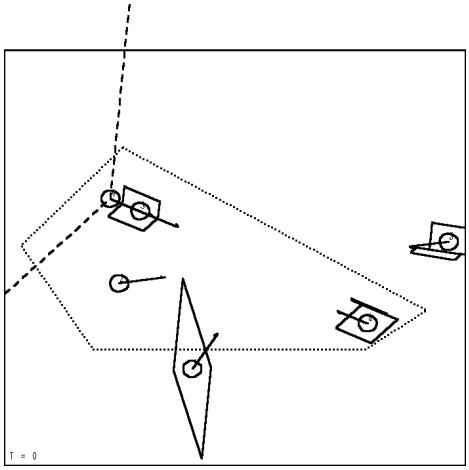

Figure 35: Conflict Parallelograms

Parallelograms are with respect to aircraft 4 where the two dashed lines (representing the "particle" lines) intersect and for circles whose radius is *doubled*.

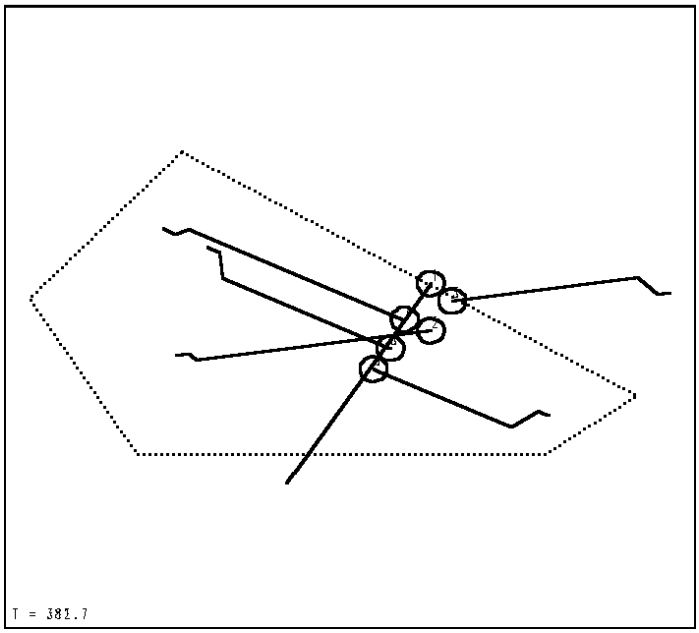

Figure 36: Three pairs of tangent circles

Resolution with equal speed parallel offset maneuvers.

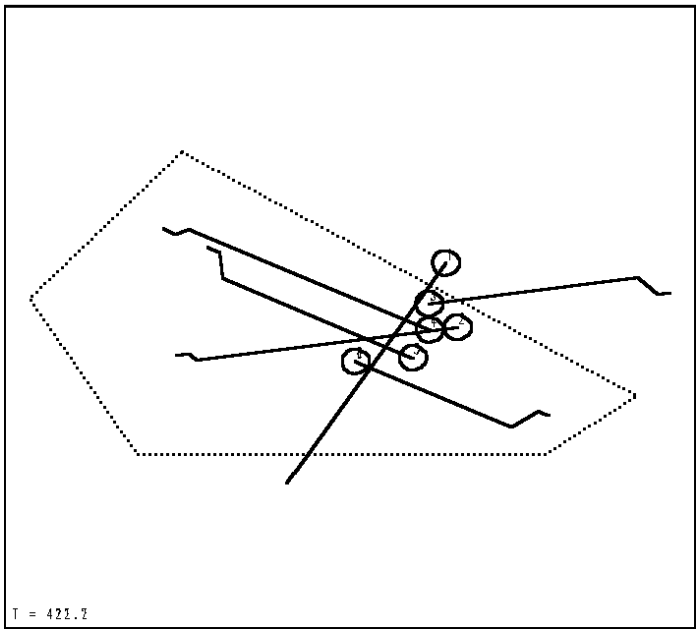

Figure 37: Triple tangency

Scraping circles indicate that the minimum displacement from original course is used.

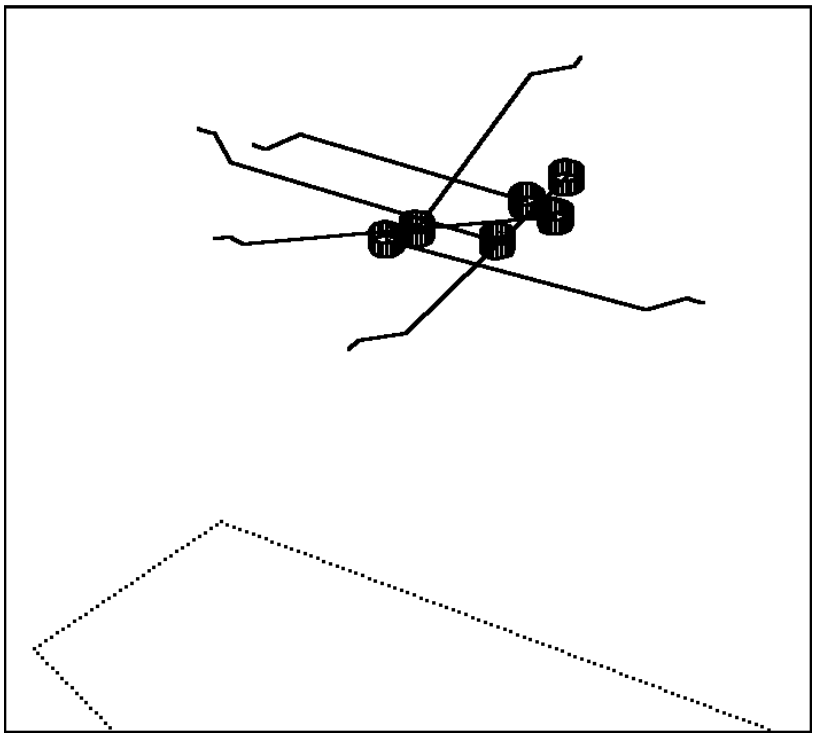

Figure 38: Resolution in 3 dimensions

Aircraft are at different altitudes and the protected space is cylindrical.

Planes, Flats & Hyperplanes

REPRESENTING HYPERPLANES WITH VERTICAL LINES

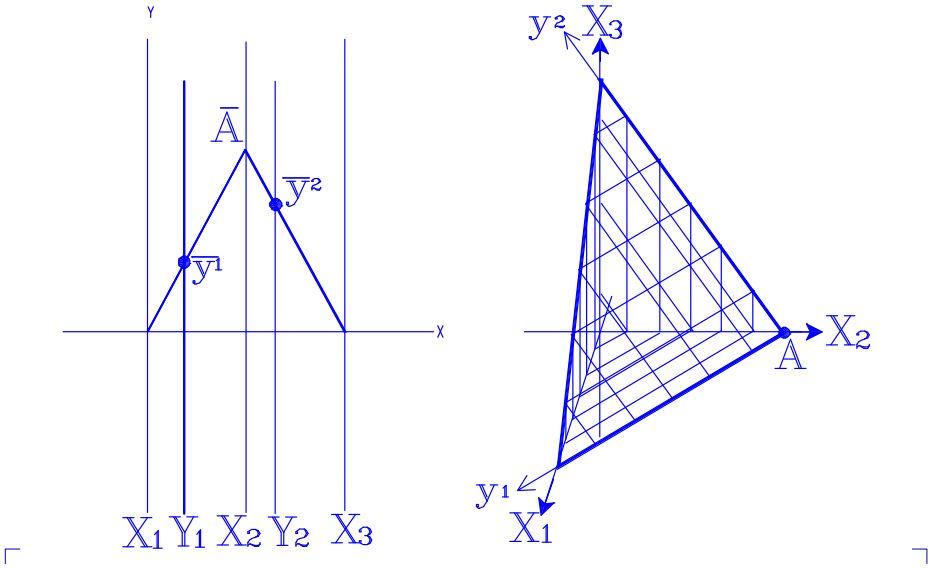

Figure 39: In P^3 planes are represented by two vertical lines and a polygonal line $\overline{A}$

This generalizes to N-dimensions where hyperplanes are represented by N-1 parallel lines and a polygonal line representing one of its points.

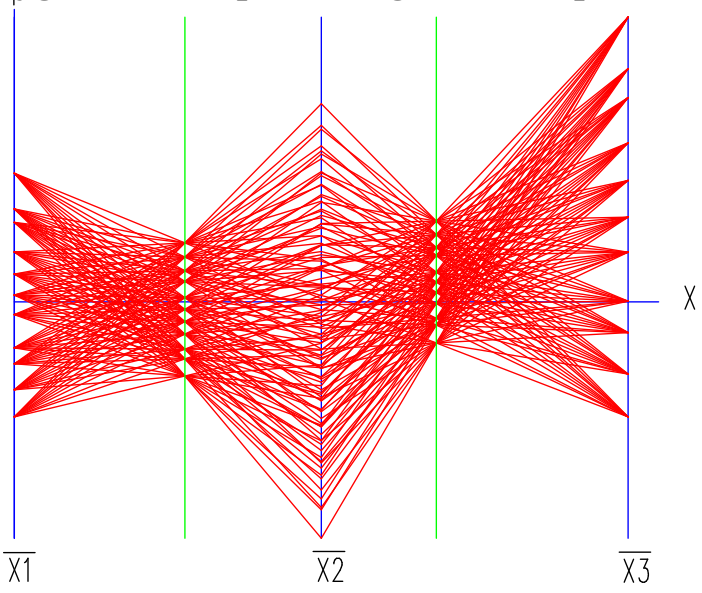

Figure 40: Set of coplanar on a regular grid points in 3-D

EXAMPLE – INDUSTRIAL DATA

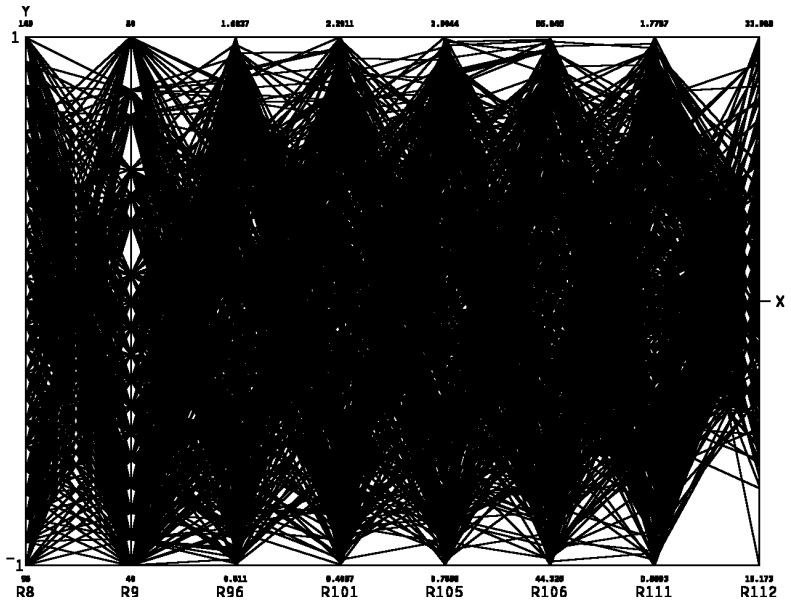

Figure 41: Industrial Data. Note pattern between the variables R111 and R112

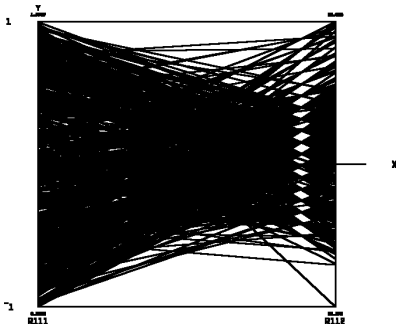

Figure 42: Enlarged R111 - R112 portion of previous plot

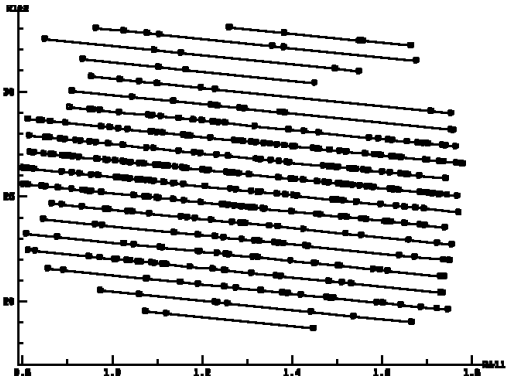

Figure 43: R111 vs. R112 linear relation between these 2 and another parameter

DUALITY BETWEEN TRANSLATIONS OF A POINT ALONG A LINE AND ROTATIONS OF A LINE ABOUT A POINT

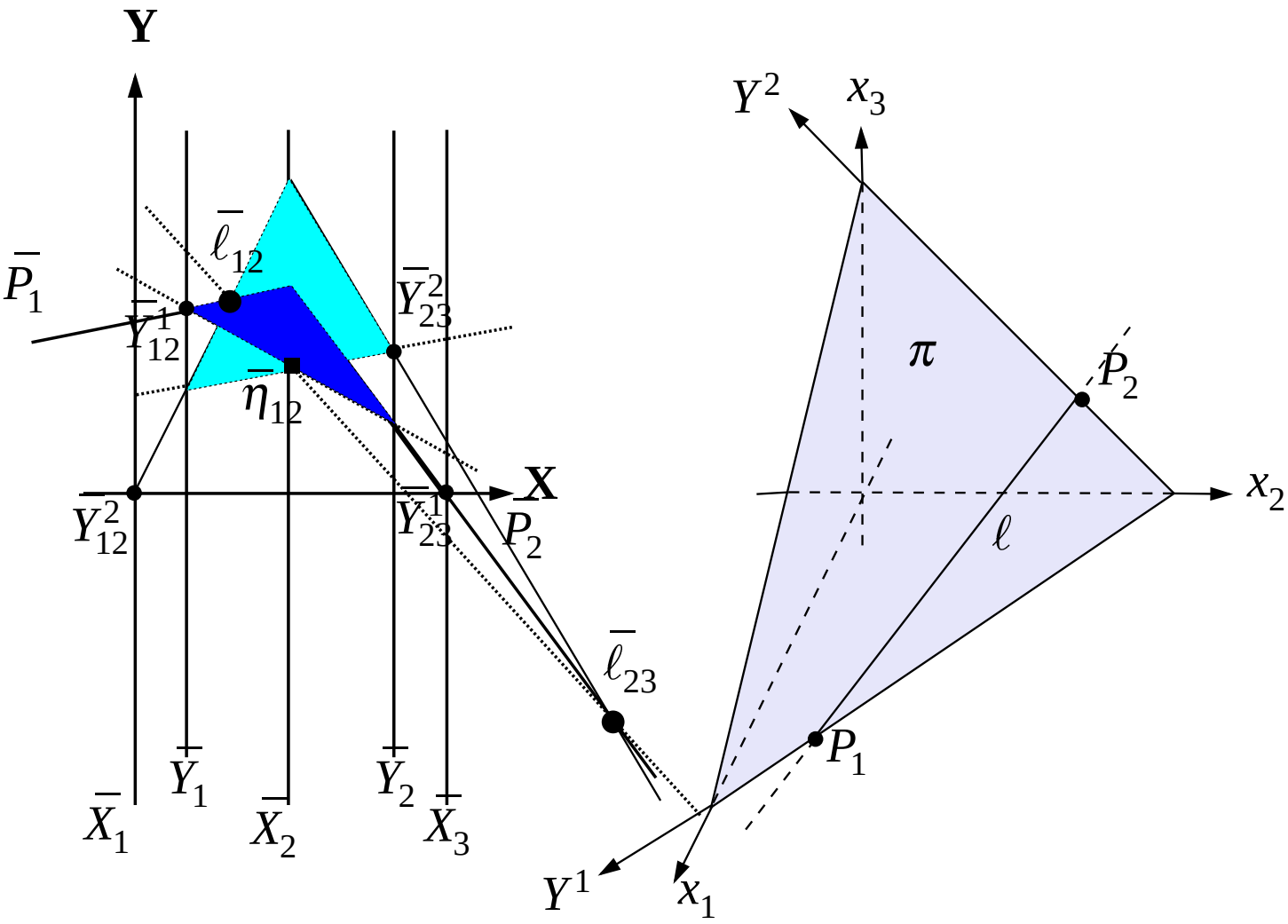

Figure 44: A line ℓ on a plane π is represented by one point $\bar{\eta}_{12}$ in terms of the planar coordinates $\bar{Y}_1$ and $\bar{Y}_2$ which is collinear with it's two point $\bar{\ell}_{12}$ and $\bar{\ell}_{23}$.

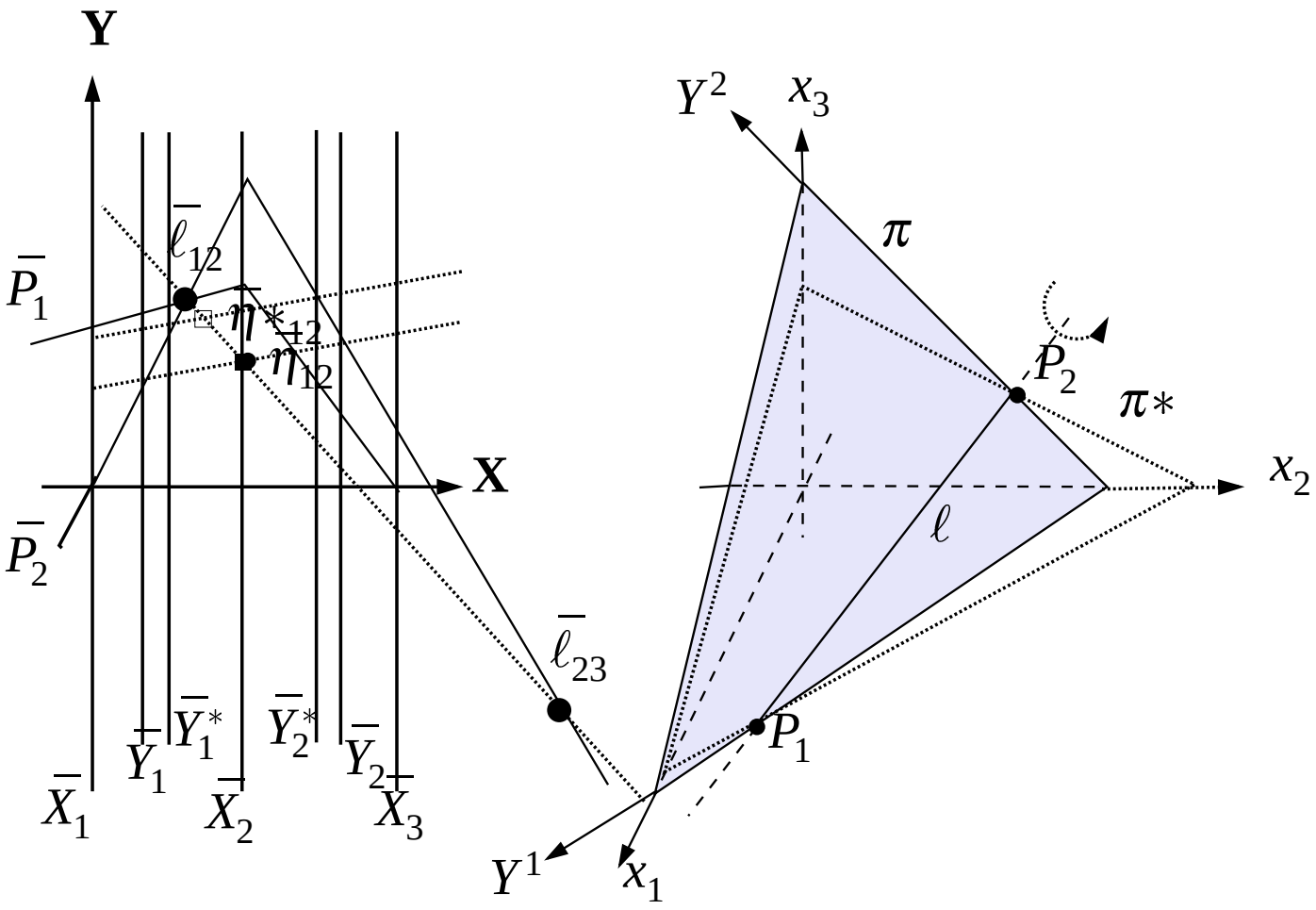

Figure 45: Rotation of a plane about a line ↔ Translation of a point along a line.

REPRESENTING FLATS BY INDEXED POINTS

From Ph. D. Thesis of J. Eickemeyer @ UCLA

♡ **A** $p-flat$ **in** R^N **specified by** $N-p$ **linearly independent equations of the form :**

♡

$$\pi_{i_1 \ldots i_{(p+1)}} \ : \ \sum_{k=1}^{p+1} c_{i_k} x_{i_k} \ = \ c_o$$

♡ **can be represented by the** $(N-p) * p$ **points :**

♡

$$\overline{\pi}_{i_1^* \ldots i_{(p+1)}^*} \ : \ (\sum_{k=1}^{p+1} d_{i_k}^* c_{i_k} \ , \ c_o \ , \ \sum_{k=1}^{p+1} c_{i_k})$$

for $O \leq p < N$

♡ **where**

1. **each variable** x_i **appears on** *two* **parallel axes** $\overline{X}_i$ **and** $\overline{X'}_i$**,**
2. d_i^* **is the distance from the y-axis to the** $\overline{X}_i^*$ **axes,**
3. **Permutations such as** $i_1 i_2 \ldots i_q$ **consists of unique integers in** $[1, 2, \ldots, N]$**,**
4. **and** $i_k^* \ = \ i_k$ **when** $j \leq k$ **or** $i_k^* \ = \ i_k'$ **for** $j > k$.

DETECTING RANDOMLY CHOSEN COPLANAR POINTS

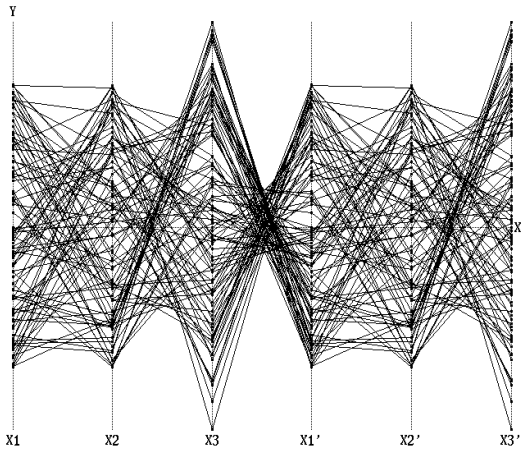

Figure 46: On the first 3 axes a set of randomly chosed coplanar points is shown

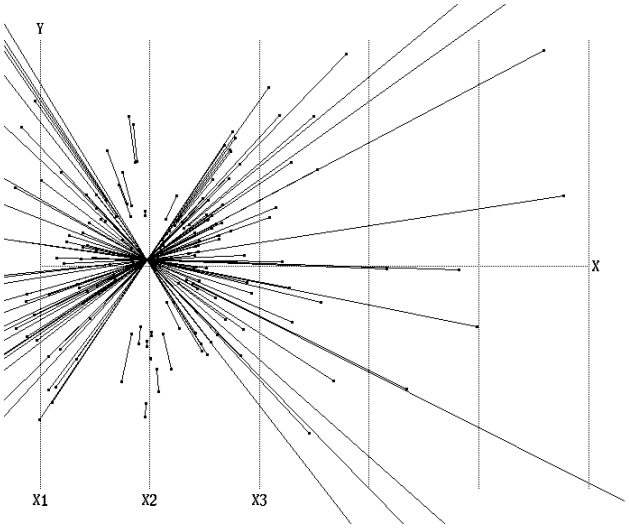

Figure 47: Coplanarity

From the points in Fig. 46 the two point representation of the lines is constructed. The lines on these points form the pencil of lines shown in Fig. 47 – this occurrs only the original points are coplanar.

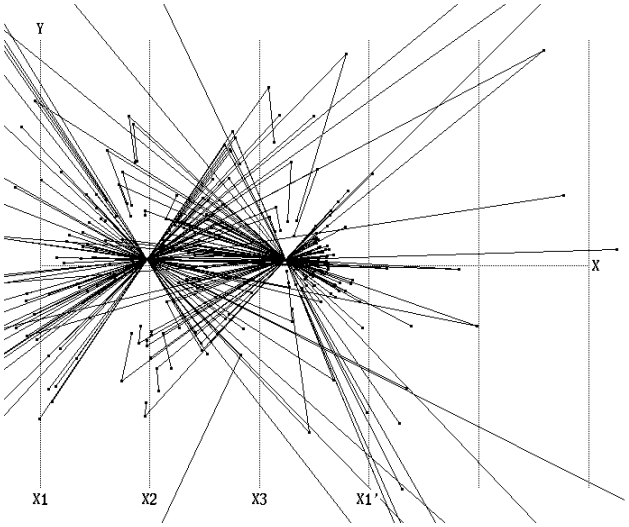

Figure 48: A plane in 3-dimensions is represented by 2 points

Second point is generated by translating the $\overline{X}_1$ to the $\overline{X}_1'$ axis and repeating the process.

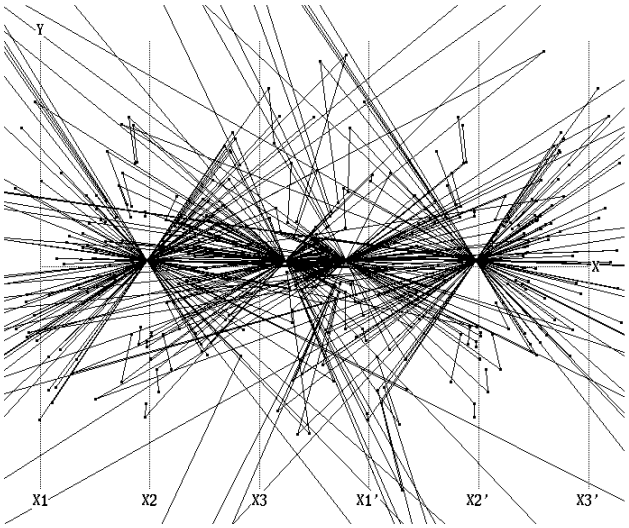

Figure 49: Four points generated from the coplanar points

The points are generated with the $\overline{X}_i'$ $(i = 1,2,3)$ axes.

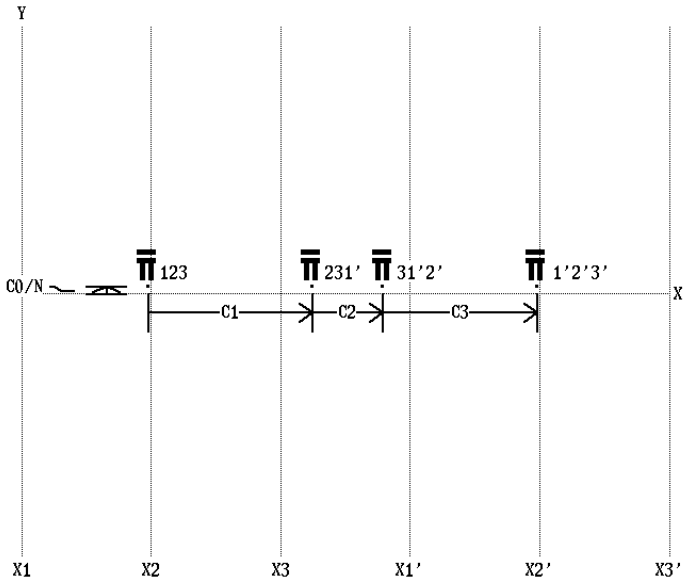

Figure 50: Reading the equation of a plane from its representation

A plane $\pi : c_1x_1 + c_2x_2 + c_3x_3 = c_o$. The coefficients are the distances between adjacent (by indices) points.

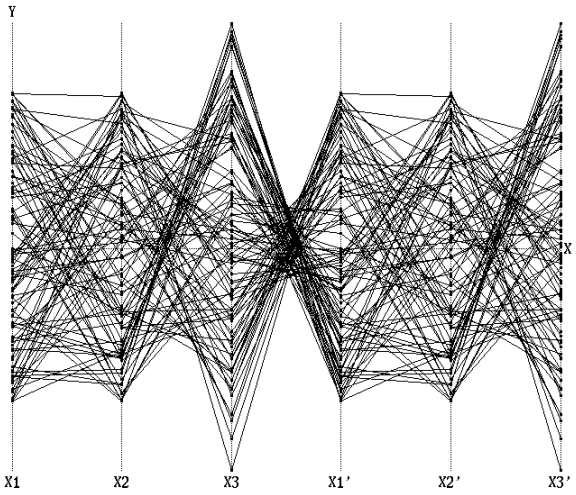

Figure 51: Randomly chosed points on an approximate plane ("slab") in 3-dimensions on left 3 axes

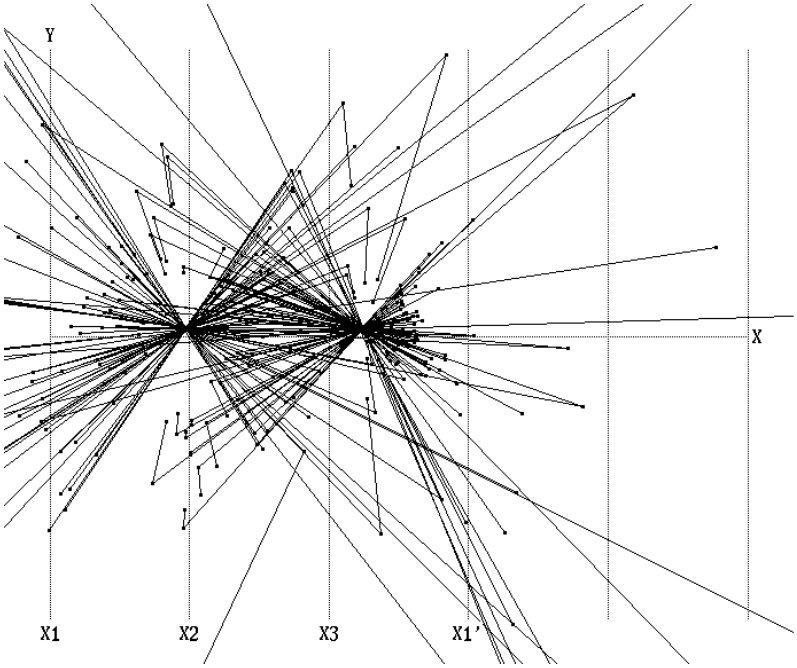

Figure 52: Approximate coplanarity obtained using the points shown in Fig. 51.

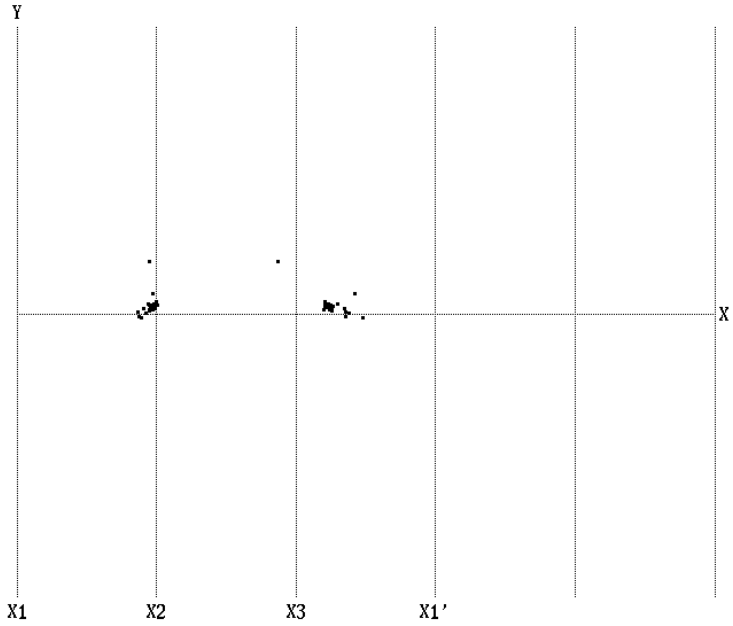

Figure 53: The point clusters indicating the approximate plane – from the points shown in Fig. 51.

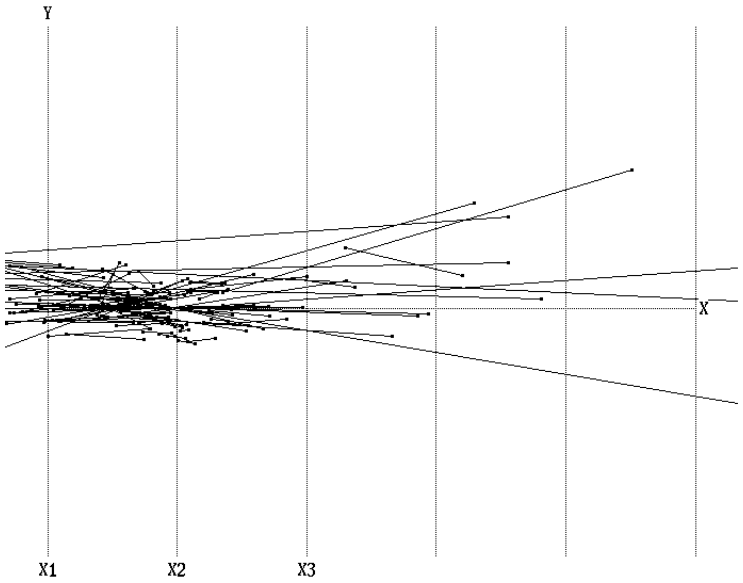

Figure 54: Detection of several approximate planes (slabs)

Starting from a set of points, represented by polygonal lines, lines are formed.

No pattern is seen since points are not from a single slab.

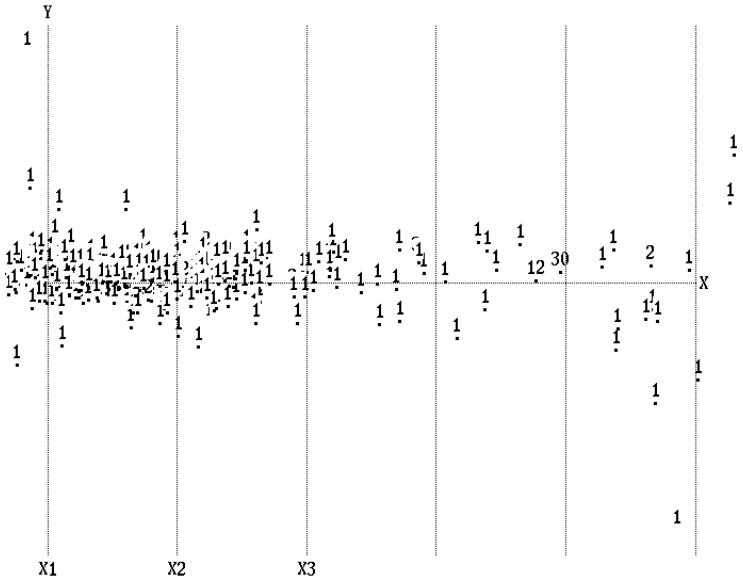

Figure 55: Detecting several slabs from randomly chosen points

A histogram giving the number of intersections per point.

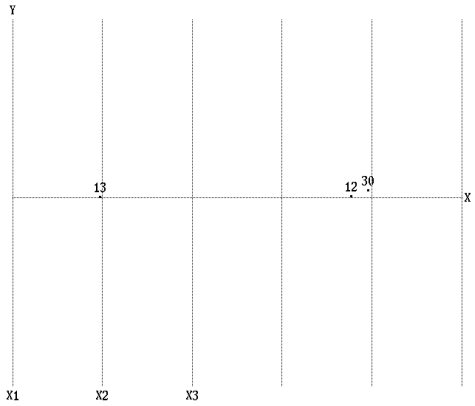

Figure 56: Original points belonged to 3 slabs

Histogram is queried for points with more than 2 hits.

HIGHER DIMENSIONAL EXAMPLES

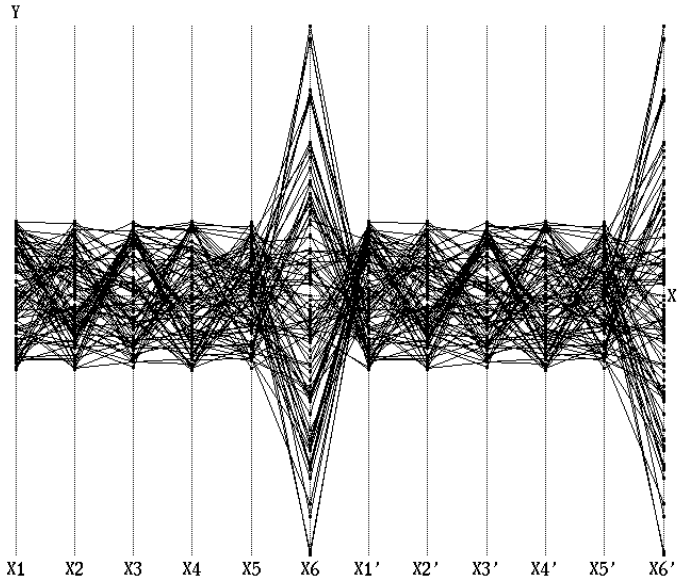

Figure 57: Points (0-flats) on an approximate hyperplane in 6-dimensions

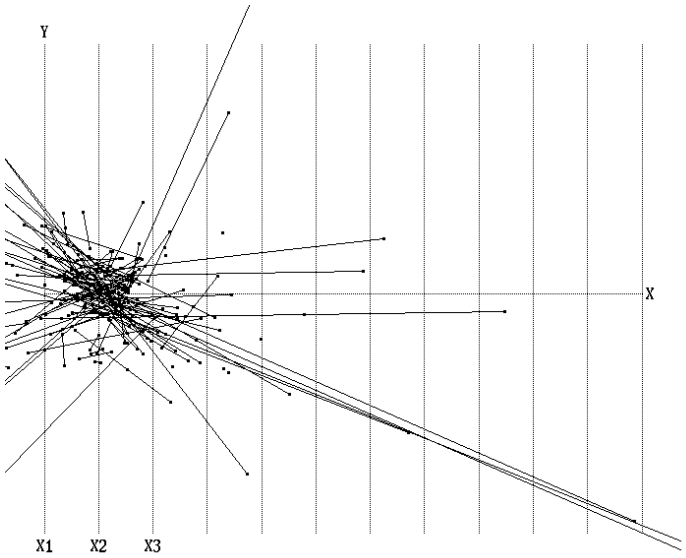

Figure 58: Portions of Lines (1-flats) formed from the previous points

No “structure” is evident

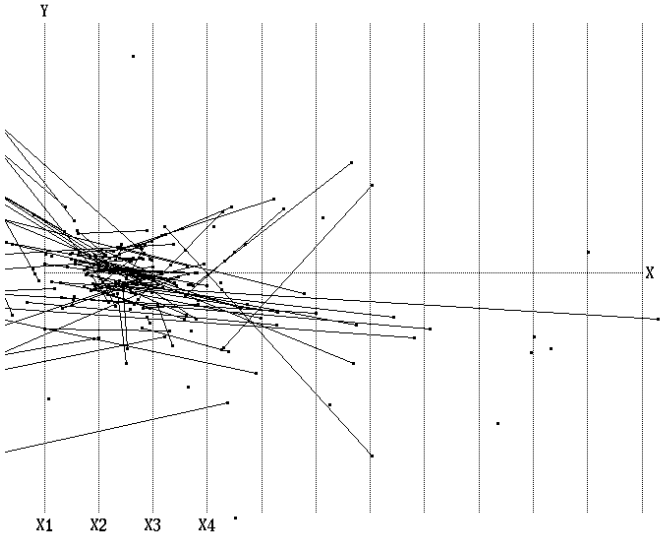

Figure 59: Portions of planes (2-flats) formed from the previous lines

Again no pattern is seen.

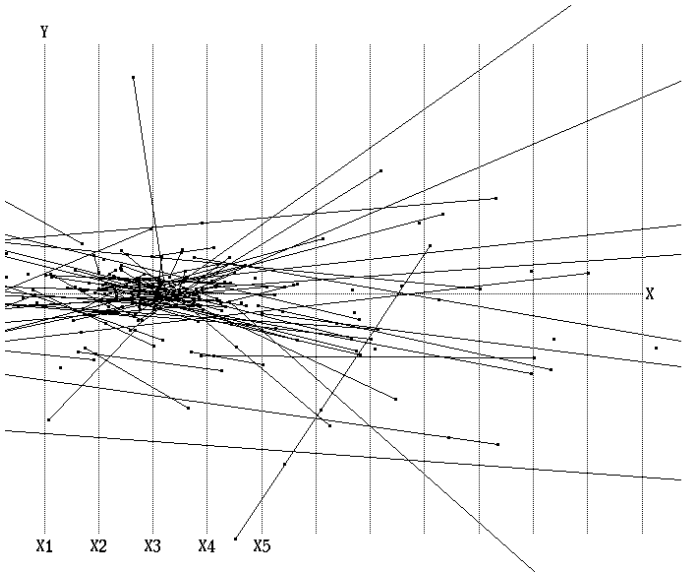

Figure 60: Portions of 3-Flats formed from the previous 2-flats

No apparent "structure"

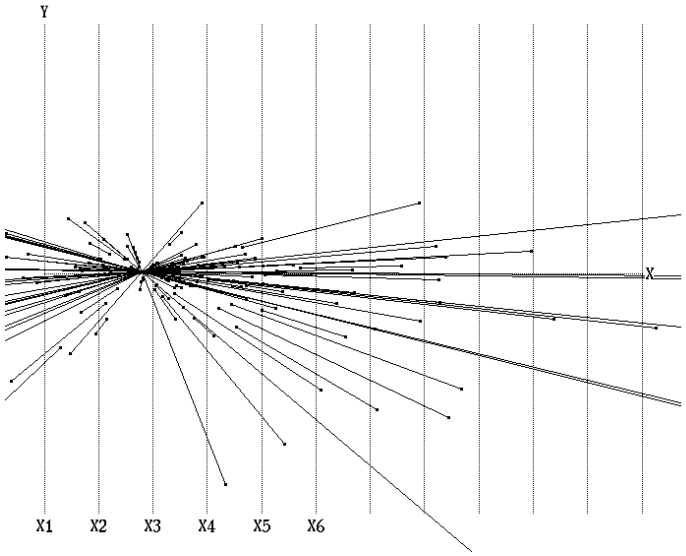

Figure 61: Portions of 4-Flats formed from the previous 3-flats

Pencil of lines showing that the original points are very near to a hyperplane (5-Flat) in 6-dimensions.

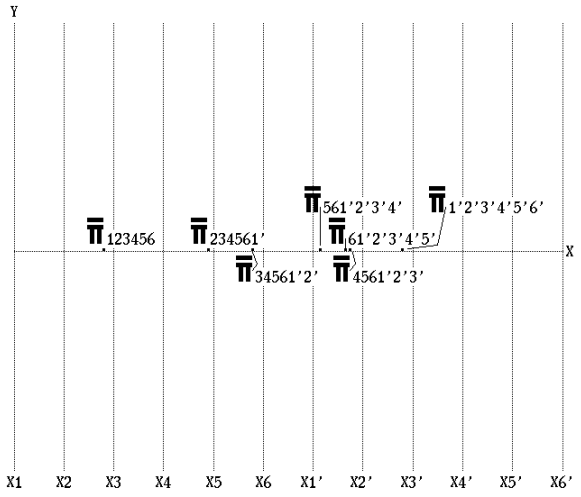

Figure 62: Points representing the hyperplane in R^6

Repeating the process in terms of the auxiliary axes $\overline{X}_i'$, $i = 1, 2, \ldots, 6$ yields the points representing the hyperplane. As in the 3-D example the distance between adjacent by index pair of points provides the coefficients of the hyperplanes equation.

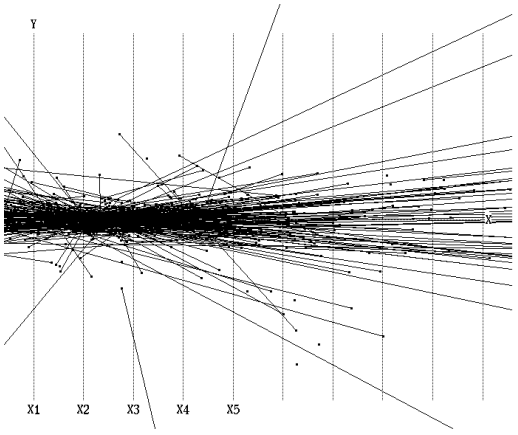

Figure 63: Detecting points belonging to several slabs in 5-D

Portions of 4-flats formed from original set of 5-D points.

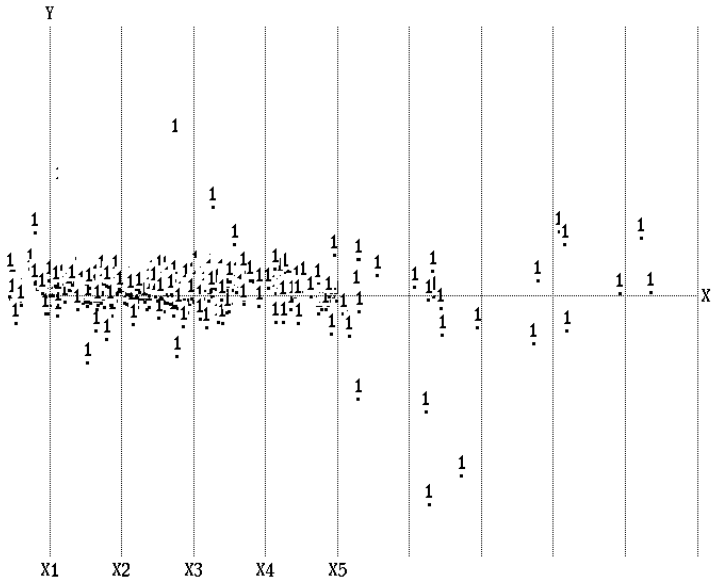

Figure 64: Number of intersections per position

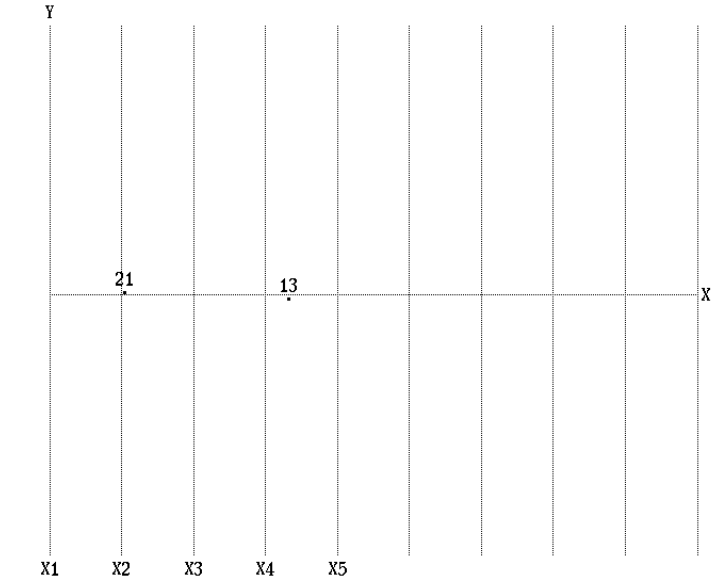

Figure 65: Two “hits” with more than 1 intersection. Points are on *two* hyperplanes

MORE ADVANCED DATAMINING

Visual Data Mining

Selected Examples – an effort will be made to match the audience's interests

A Geometric Classifier

Classification is a basic task in data analysis and pattern recognition and an algorithm accomplishing it is called a Classifier. The input is a dataset P and a designated subset S. The output is a characterization, that is a set of conditions or rules, to distinguish elements of S from all other members of P. With parallel coordinates a dataset P with N variables is transformed into a set of points in N-dimensional space. In this setting, the designated subset S can be described by means of a hypersurface which encloses just the points of S. In practical situations the strict enclosure requirement is dropped and some points of S may be omitted ("false negatives"), and some points of $P - S$ are allowed ("false positives") in the hypersurface. The description of such a hypersurface is equivalent to the rule for identifying, within some acceptable error, the elements of S. This is the *geometrical* basis for the classifier presented here. The algorithm accomplishing this entails:

♢ **use of an efficient "wrapping" algorithm to enclose the points of S in a hypersurface S_1 containing S and typically also some points of $P-S$; so $S \subset S_1$, of course such an S_1 is not unique.**

the points in $(P-S) \cap S_1$ are isolated and the wrapping algorithm is applied to enclose them, and usually also a few points of S_1, producing a new hypersurface S_2 with $S \supset (S_1 - S_2)$,

♢ **the points in S not included in $S_1 - S_2$ are next marked for input to the wrapping algorithm, a new hypersurface S_3 is produced containing these points as well as some other points in $P - (S_1 - S_2)$ resulting in $S \subset (S_1 - S_2) \cup S_3$,**

♢ **the process is repeated alternatively producing upper and lower containment bounds for S; termination occurs when an error criterion (which can be user specified) is satisfied or when convergence is not achieved.**

It can and does happen that the process does not converge when P does not contain sufficient information to characterize S. It may also happen that S is so "porous" (i.e. sponge-like) that an inordinate number of iterations are required. On convergence the output is a description of the hypersurface containing S the rule is given in terms of the minimum number of variables needed to describe S *without loss of information*. Unlike other methods, like the Principal Component

Analysis (PCA), the classifier discards only the redundant variables. It is important to clarify this point. A subset S of a multidimensional set P is not necessarily of the same dimensionality as P. So the classifier finds the dimensionality of S in terms of the original variables and retains only those describing S. That is, it finds the *basis* in the mathematical sense of the smallest subspace containing S, or more precisely the current approximation for it. This basis is the minimal set M_r of variables needed to describe S. We call this dimensionality selection to distinguish it from dimensionality *reduction* which is usually done *with* loss of information. Retaining the original variables is important in the applications where the domain experts have developed intuition about the variables they measure. The classifier presents M_r *ordered according to a criterion which optimizes the clarity of separation*. This may be appreciated with the example provided in the attached figure, in addition.

The implementation allows the user to select a subset of the available variables and restrict the rule generation to these variables. In certain applications, as in process control, not all variables can be controlled and hence it would be useful to have a rule involving such variables that are "accessible" in a meaningful way. There are also two options available :

- **either minimize the number of variables used in the rule, or**

- **minimize the number of steps, in terms of the unions and (relative) complements, in the rule.**

The classifier provides:

- **an approximate convex-hull boundary for each cavity is obtained,**
- **utilizing properties of the representation of multidimensional objects in $\|$-coords, a very low polynomial worst case complexity of $O(N^2|P|^2)$ in the number of variables N and dataset size $|P|$ is obtained; it is worth contrasting this with the often unknown, or unstated, or very high (even exponential) complexity of other classifiers,**
- **an intriguing prospect, due to the low complexity, is that the rule can be derived in near real-time making the classifier adaptive to changing conditions,**
- **the minimal subset of variables needed for classification is found,**
- **the rule is given explicitly in terms of conditions on these variables, i.e. included and excluded intervals, and provides "a picture" showing the complex distributions with regions where there is data and "holes" with no data; that can provide significant insights to the domain experts,**

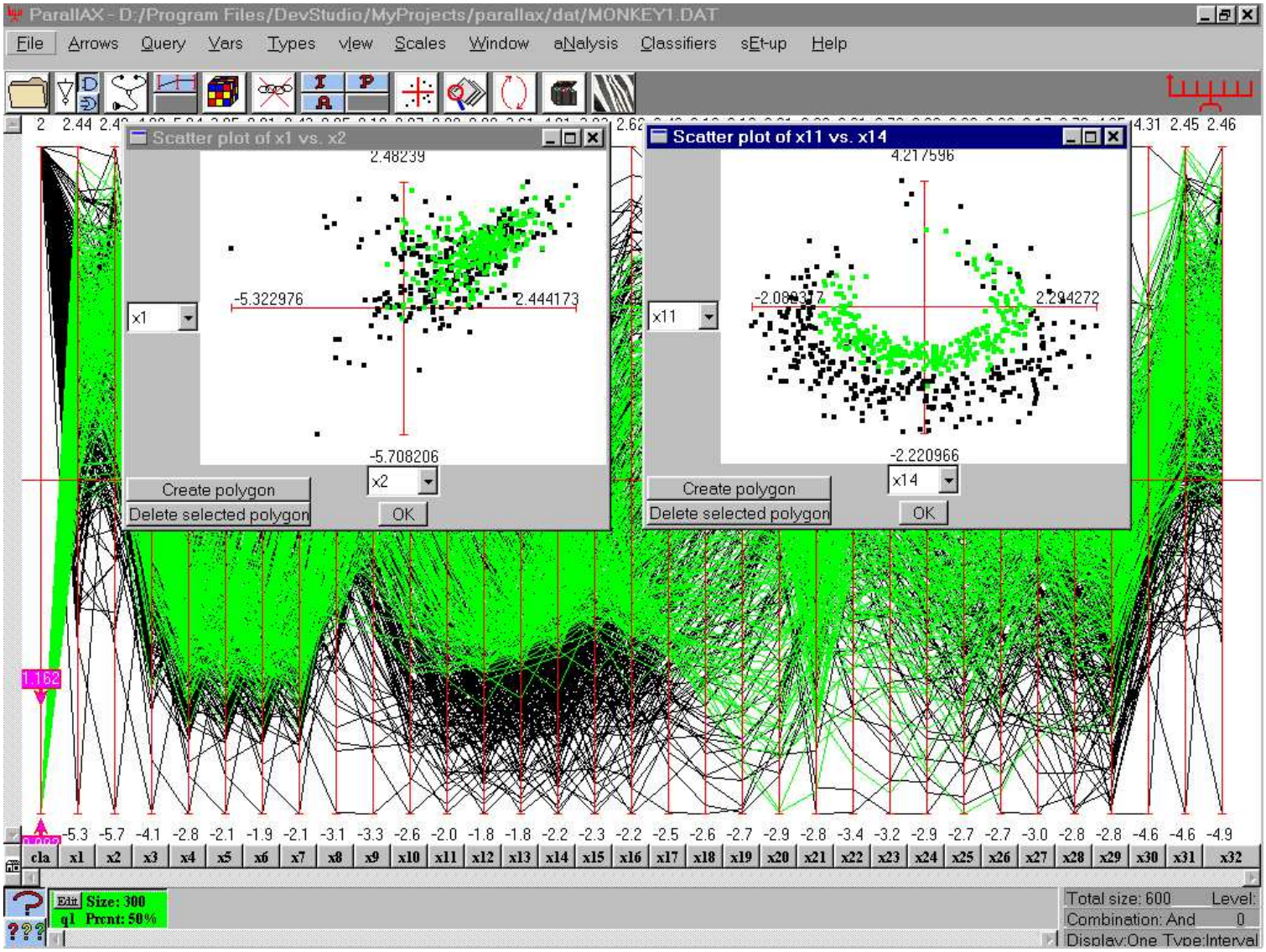

Figure 66: The monkey dataset showing the separation achieved by two of the 9 out 32 parameters obtained from the dimensionality selection.

The dataset chosen to illustrate has two classes to be distinguished consisting of pulses measured on two types of neurons in a monkey's brain (poor thing!). There are 600 samples with 32 variables. Remarkably, convergence was obtained and

required only 9 of the 32 parameters. The resulting ordering shows a striking separation. In the attached figure the first pair of variables x_1, x_2 originally given is plotted showing no separation. In the adjoining plot the best pair x_{11}, x_{14}, as chosen by the classifier's ordering, shows remarkable separation. The result shows that the data consists of two "banana-like"[1] clusters in 9-D one (the complement in this case) enclosing the other (class for which the rule was found). Note that the classifier can actually describe highly complex regions. It can build and "carve" the cavity shown. It is no wonder that separation attempts in terms of hyperplanes or nearest-neighbor techniques can fail badly on such datasets. The rule gave an error of 3.92 % using train-and-test with 66 % of the data for training).

[1] Perhaps the monkey was dreaming about bananas during this fateful experiment ...

CURVES

point-curve $\mapsto$ ***line-curve*****– envelope of it's tangents.**

Conics map into conics in six different ways.

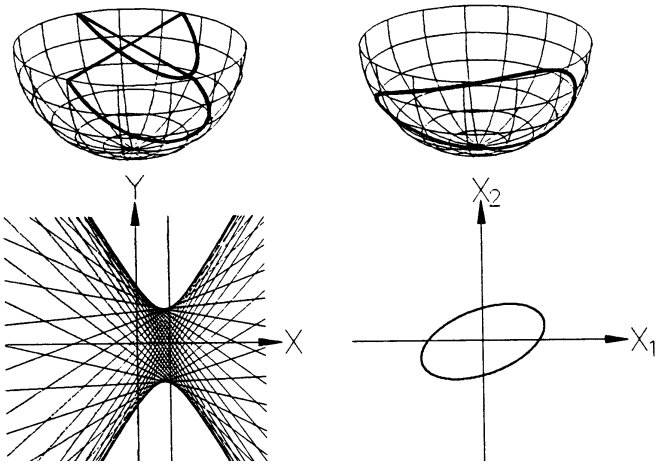

Figure 67: Ellipses always map into hyperbolas. Each assymptote is the image of a point where the tangent has slope 1.

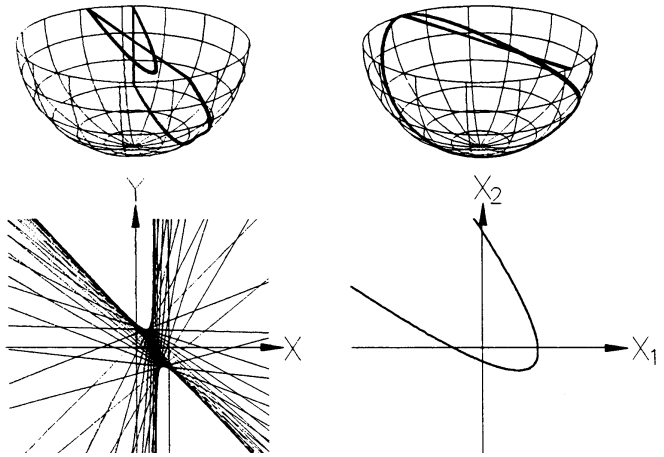

Figure 68: A parabola whose ideal point does not have direction with slope 1 **always** transforms to a hyperbola with a vertical assymptote. The other assymptote is the image of the point where the parabola has tangent with slope 1.

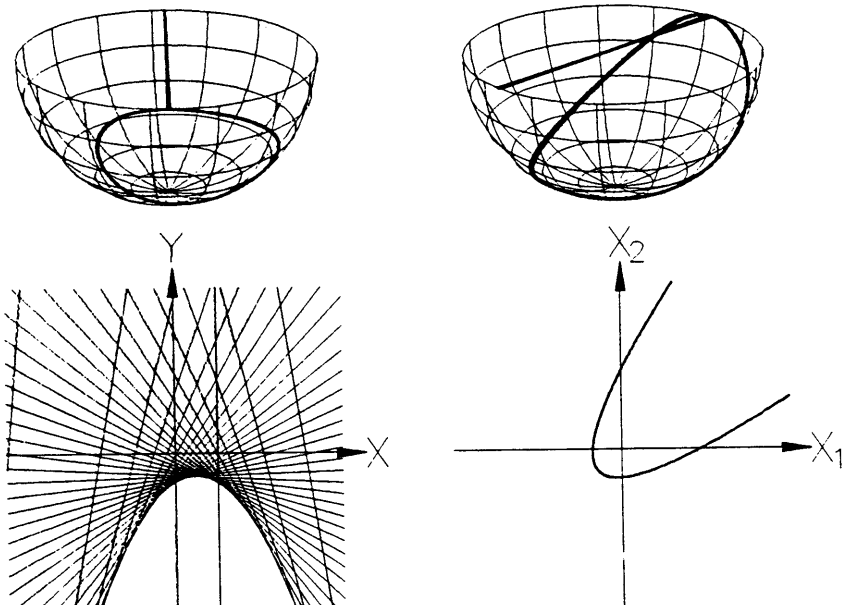

Figure 69: A parabola whose ideal point has direction with slope 1 transforms to a parabola - self-dual.

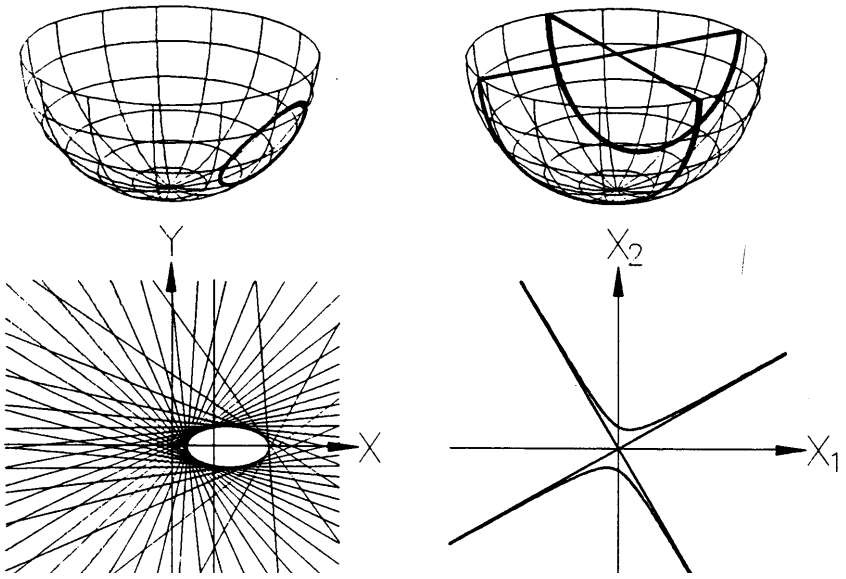

Figure 70: Hyperbola to ellipse – dual of case shown in Fig. 67

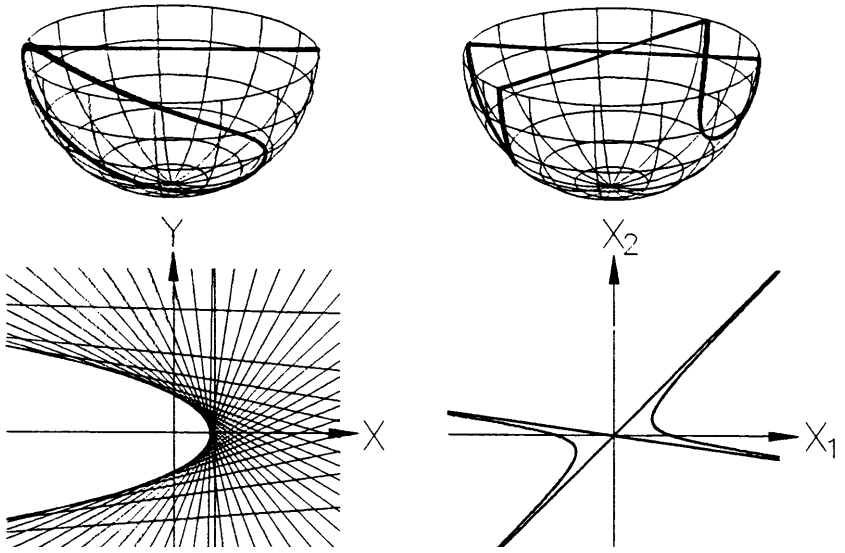

Figure 71: Hyperbola to parabola. This occurs when one of the assymptotes has slope 1 – dual of case shown in Fig. 68

.

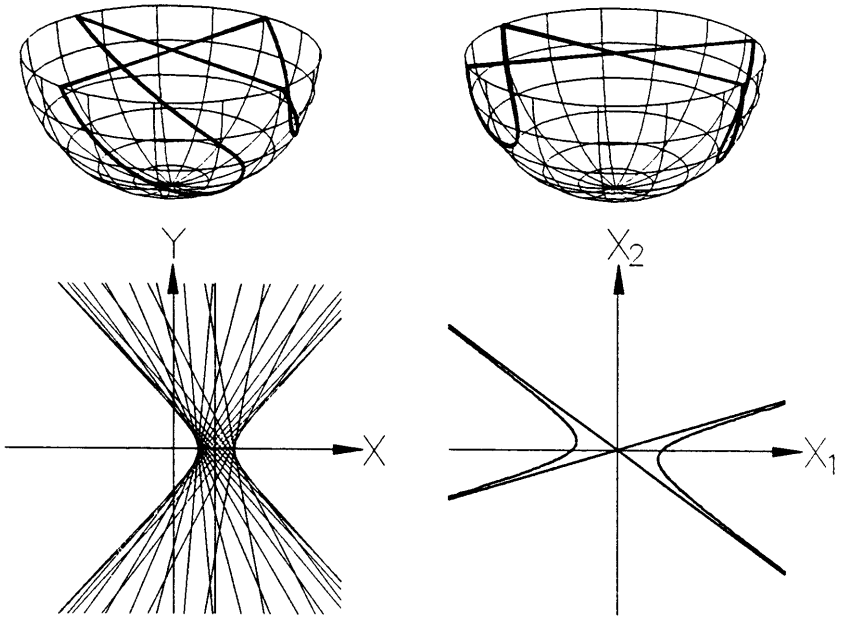

Figure 72: Hyperbola to hyperpola – self-dual case.

From M.Sc. Thesis Tsur Itshakian, CS Dept. Tel Aviv Univ. 2001

Degree $n \mapsto n(n-1)$ and less when there are singularities. An efficient algorithm was found which gives the exact equation of the image even for implicitly defined polynomials. Here a mapping *point-curve* $\mapsto$ *point-curve* is used which overcomes the over-plotting problem.

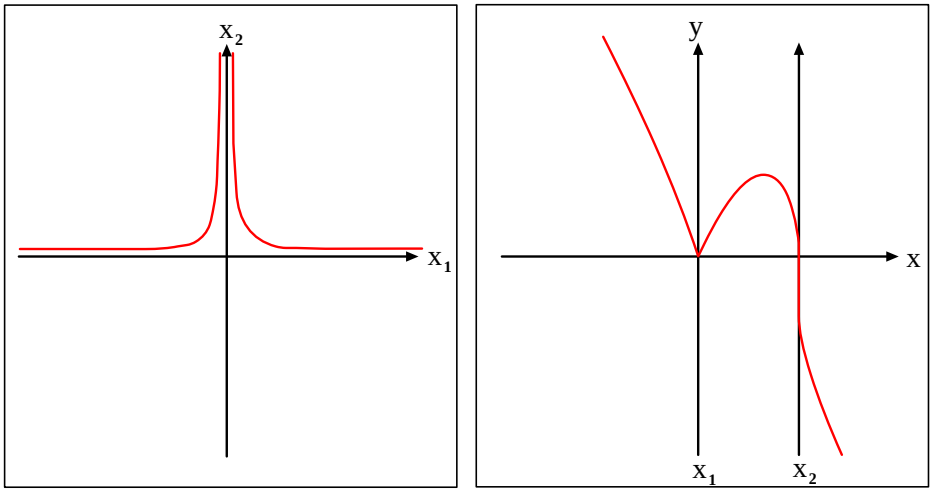

Figure 73: A 3rd degree curve with singularity maps to another 3rd degree curve.

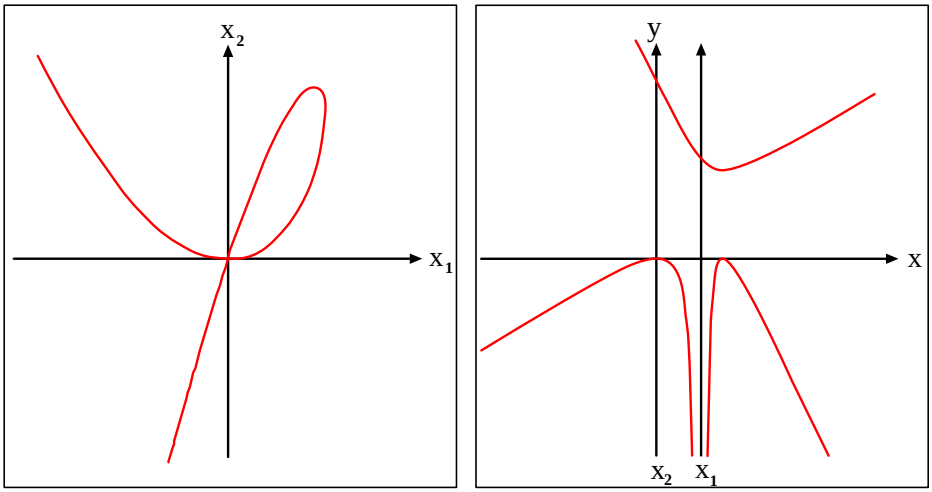

Figure 74: A 3rd degree curve with different singularity maps into a 4th degree curve.

Generalized conics – Gconics

Like conics, gconics map into gconics in 6 different ways

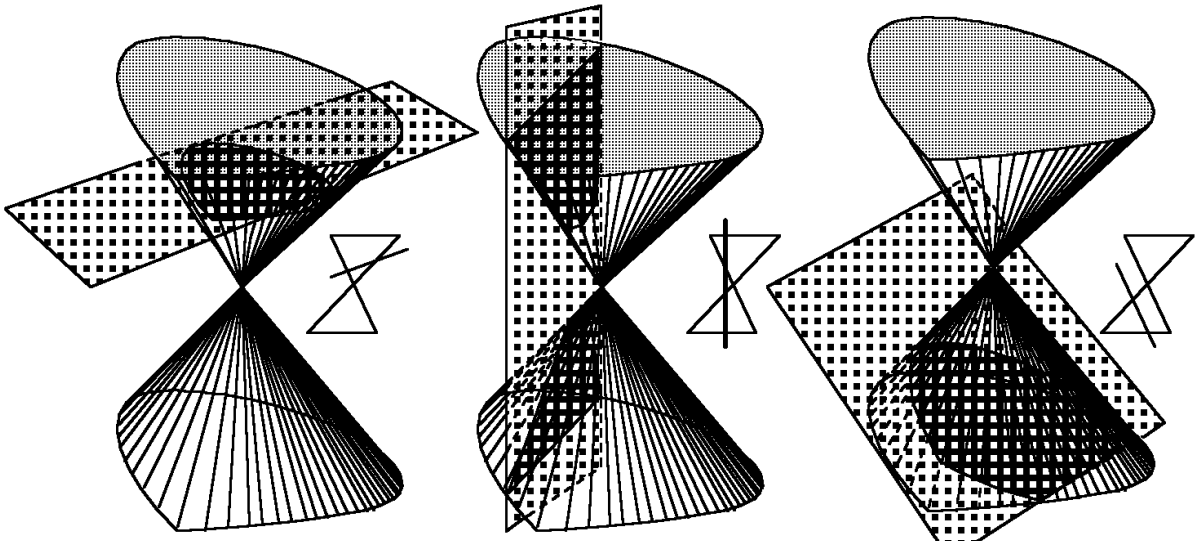

Figure 75: Gconics - three types of sections: (left) bounded convex set *bc*, (right) unbounded convex set *uc* and (middle) hyperbola-like *gh* regions.

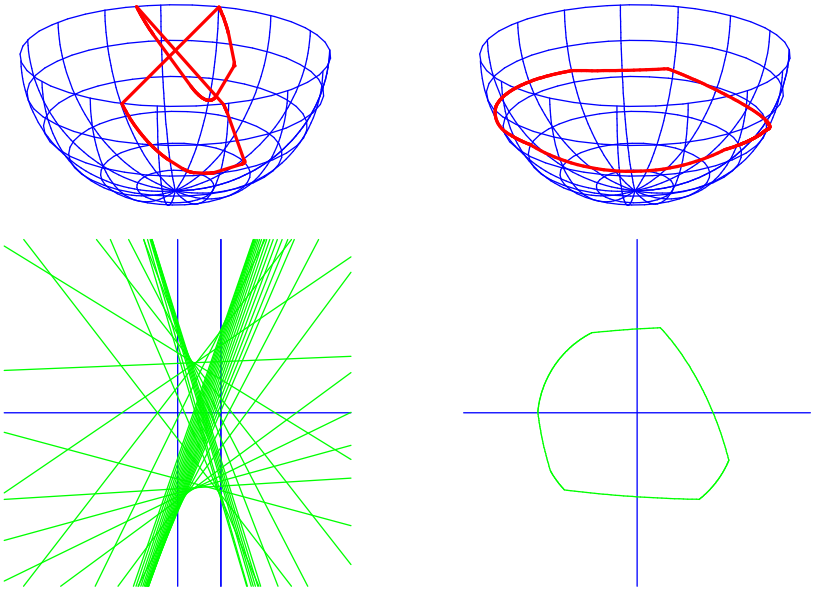

Figure 76: Generalization of Fig. 67 — *bc* to *gh*.

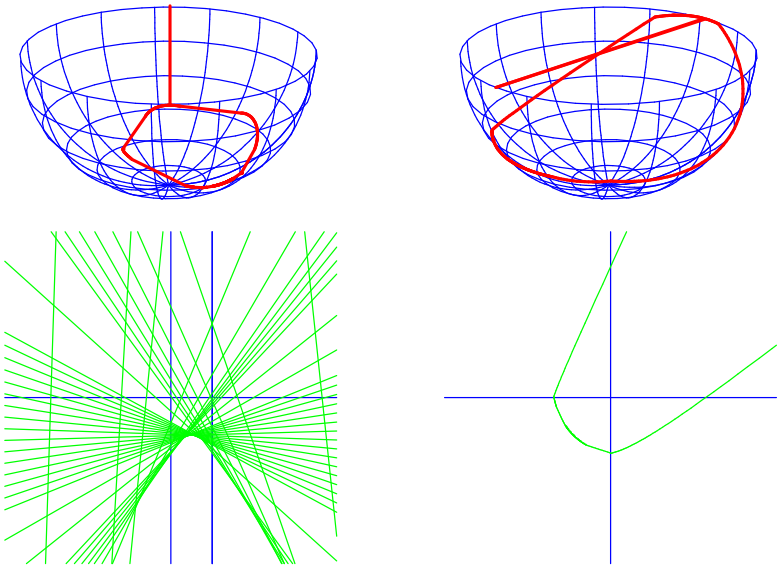

Figure 77: *uc* to *uc* – self-dual

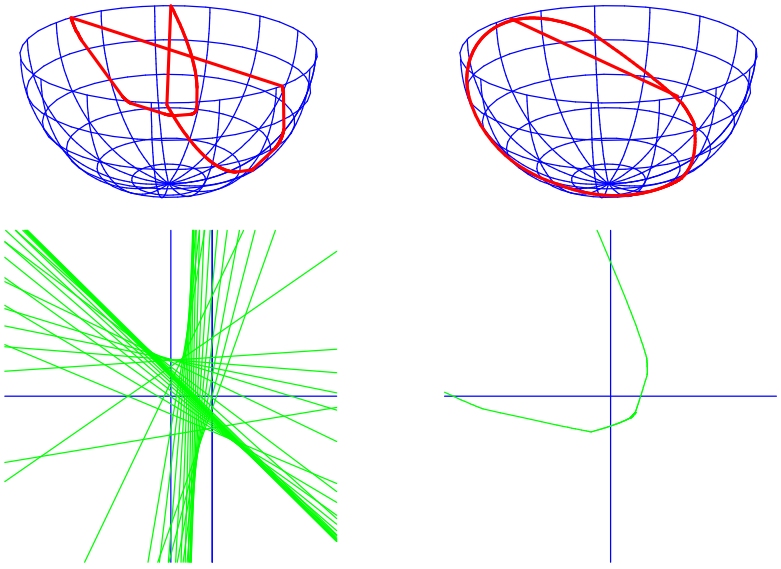

Figure 78: *uc* to *gh*

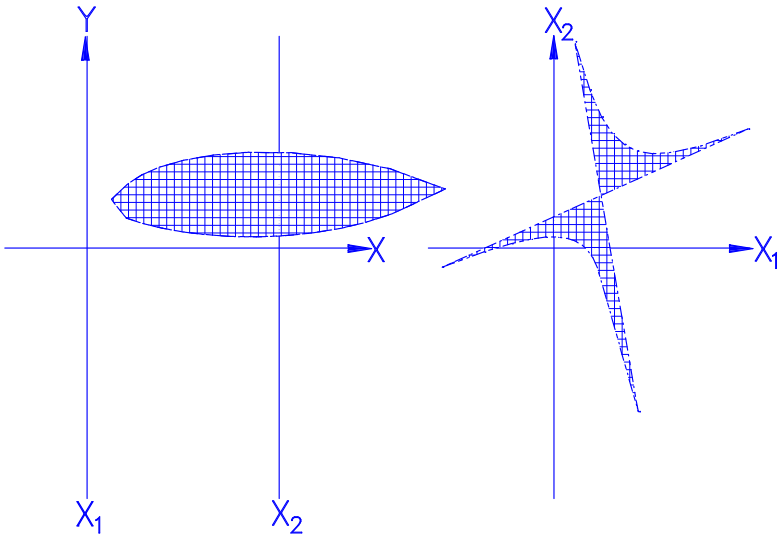

Figure 79: *gh* to *bc*.

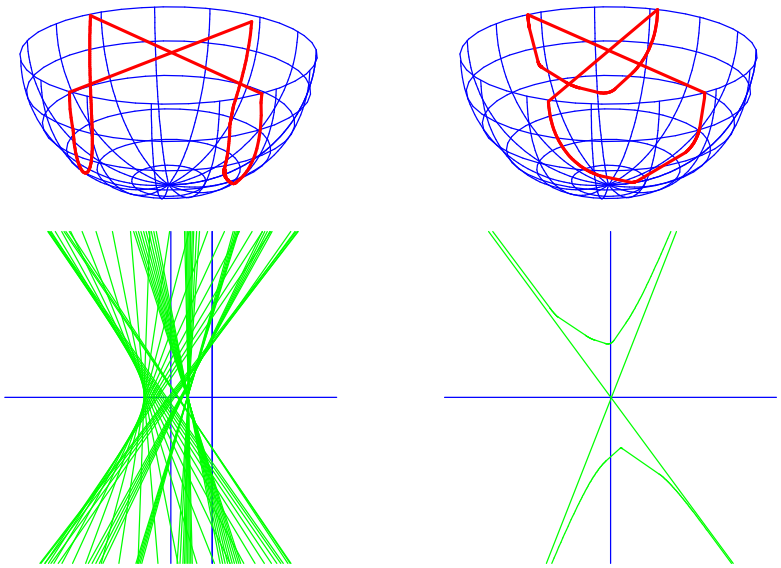

Figure 80: *gh* to *gh* – self-dual

Further Dualities

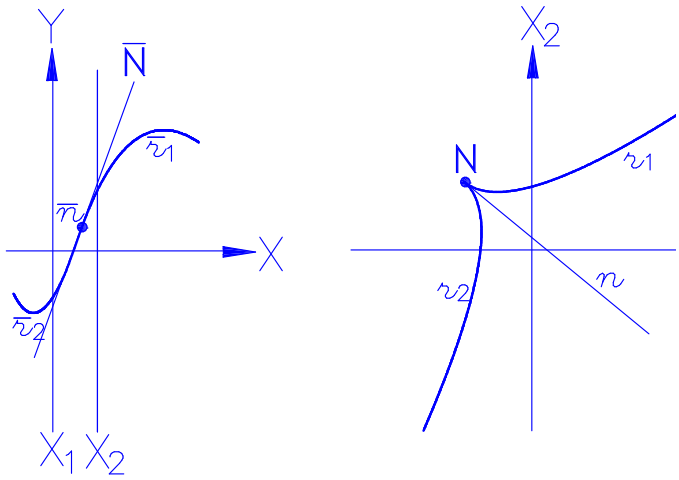

Figure 81: Cusps are transformed into inflection points

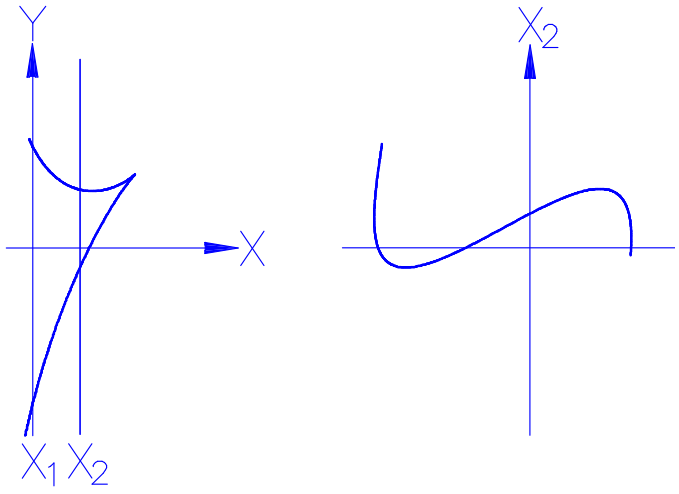

Figure 82: Duality *Cusps* $\leftrightarrow$ *Inflection Points*

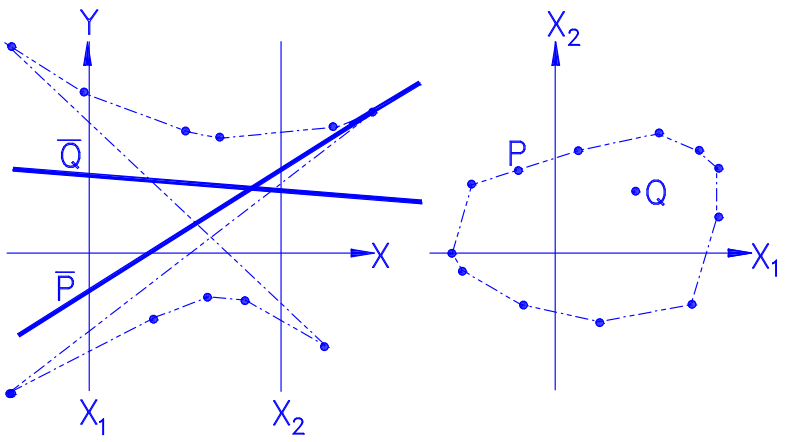

Figure 83: Interior and boundary points of bounded convex set

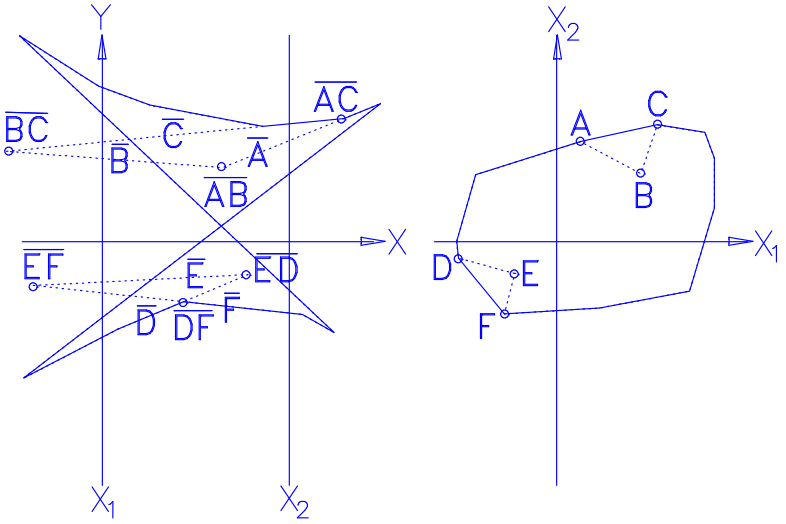

Figure 84: Convex-Hull construction

The boundary of the *gh* of a set of points corresponds to it's CONVEX-HULL.

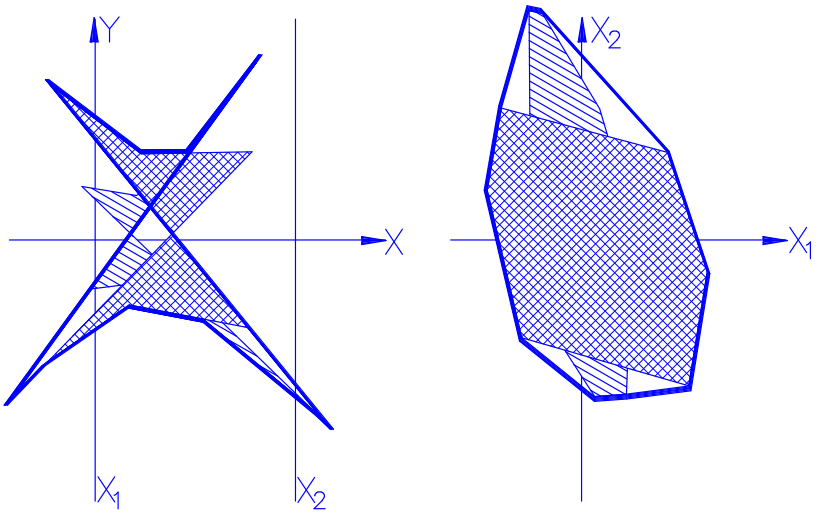

Figure 85: Convex Union of *bc*s corresponds to the Outer Union of their images *gh*s.

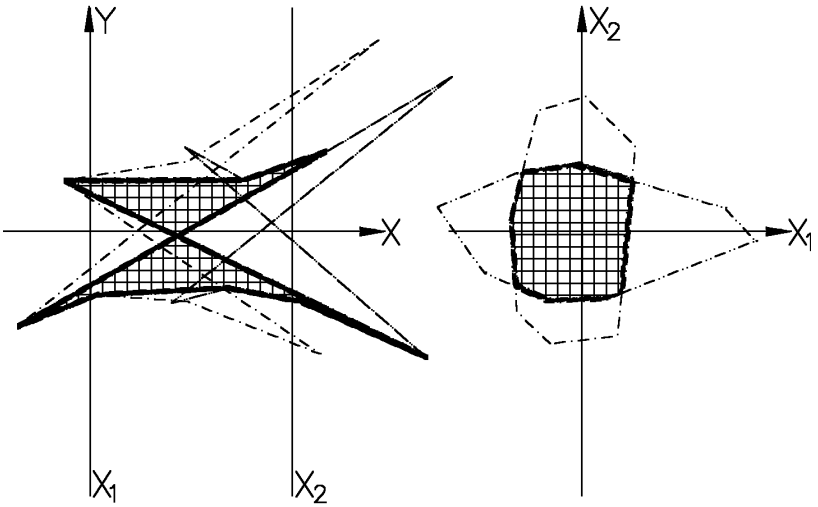

Figure 86: Inner Intersection and Intersection are Dual.

LINE NEIGHBORHOODS

A Topologoly for proximity of flats

How can measure “closeness” between lines and more general between planes?

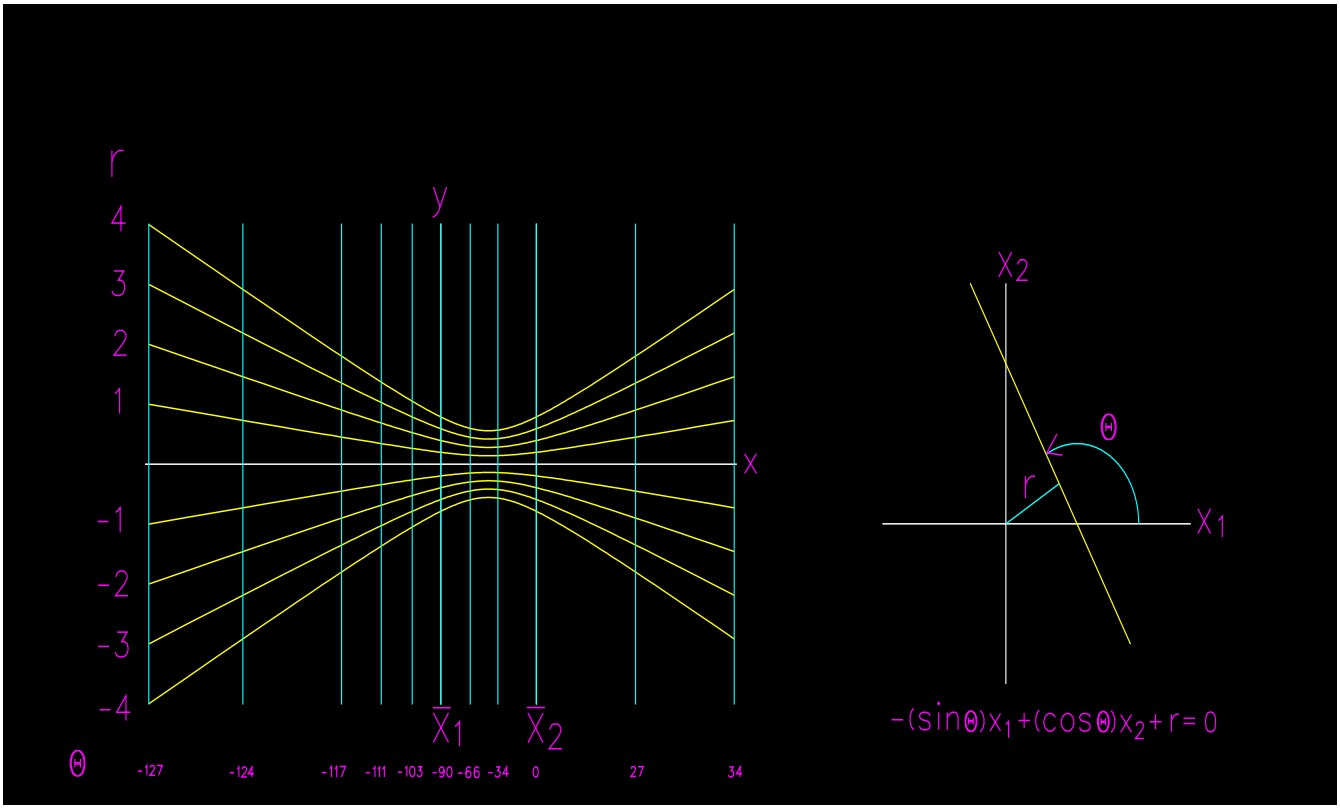

Figure 87: A family of line transformations

Fixing *r* and varying Θ defines a family of lines tangent to the circle whose parallel coordinate representation is a hyperbola while fixing Θ and varying *r* produces vertical lines.

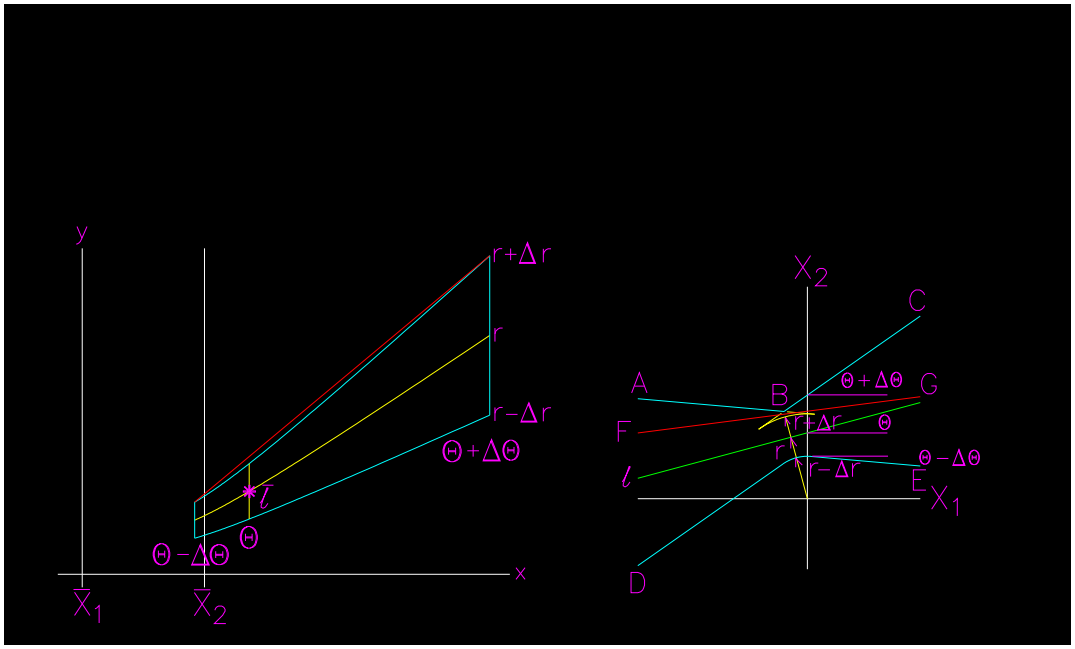

Figure 88: Line neighborhood in orthogonal(doesn't work) and parallel coordinates. The unbounded region (on the right) is replaced by a bounded one.

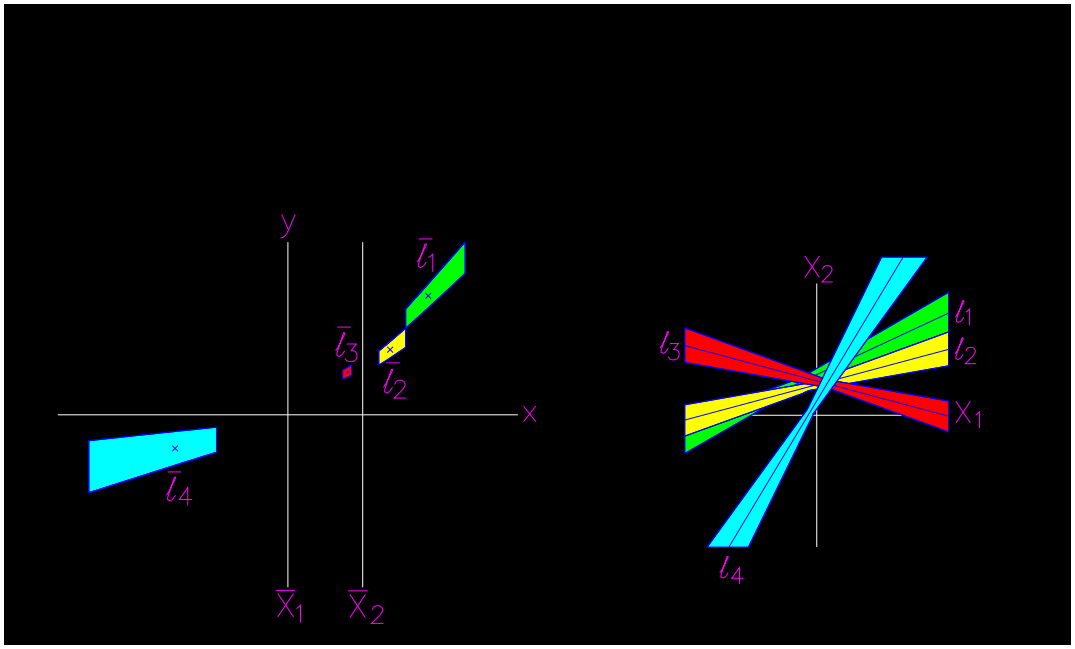

Figure 89: Several line neighborhoods. Here the transformed neighborhoods are distinct.

HYPERSURFACES

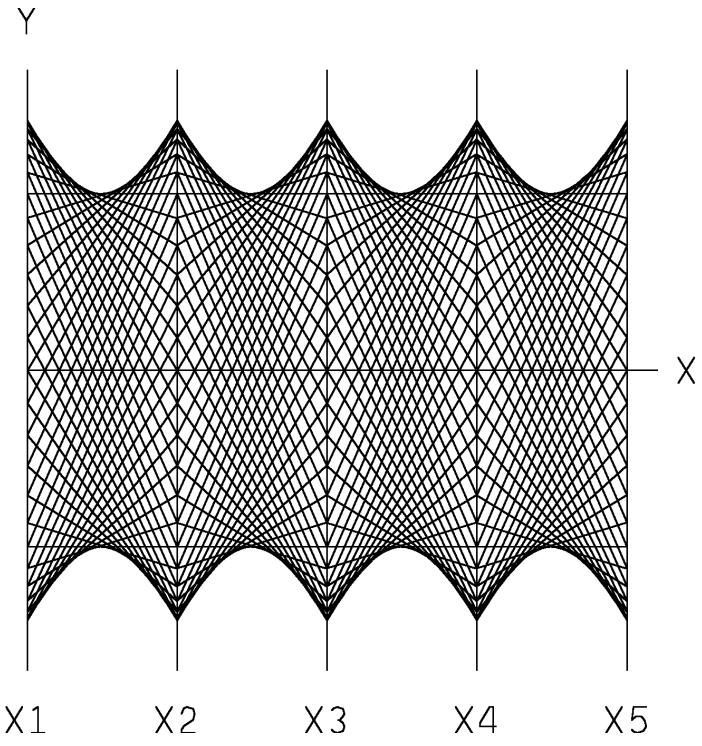

Figure 90: A sphere in R^5 centered at the origin (0,0,0,0,0).

Interior Point Construction Algorithm

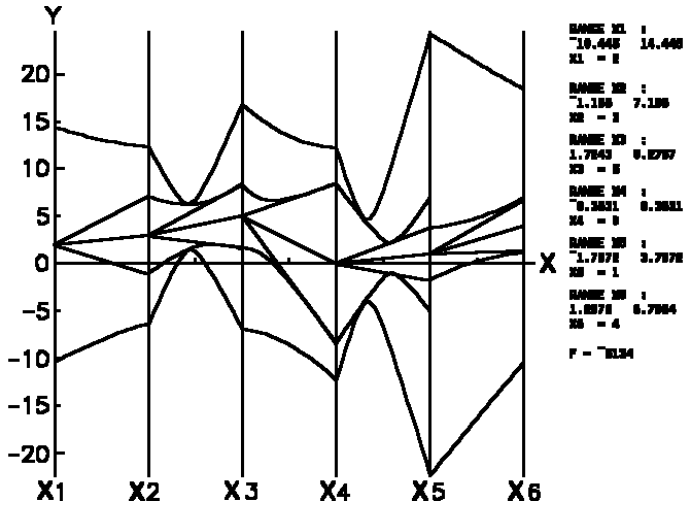

Figure 91: The polygonal line represents the point found interior to the Hyperellipsoid in 6-D. The same algorithm applies to any piecewise convex hypersurface.

Application to Process Control and Intelligent Instrumentation

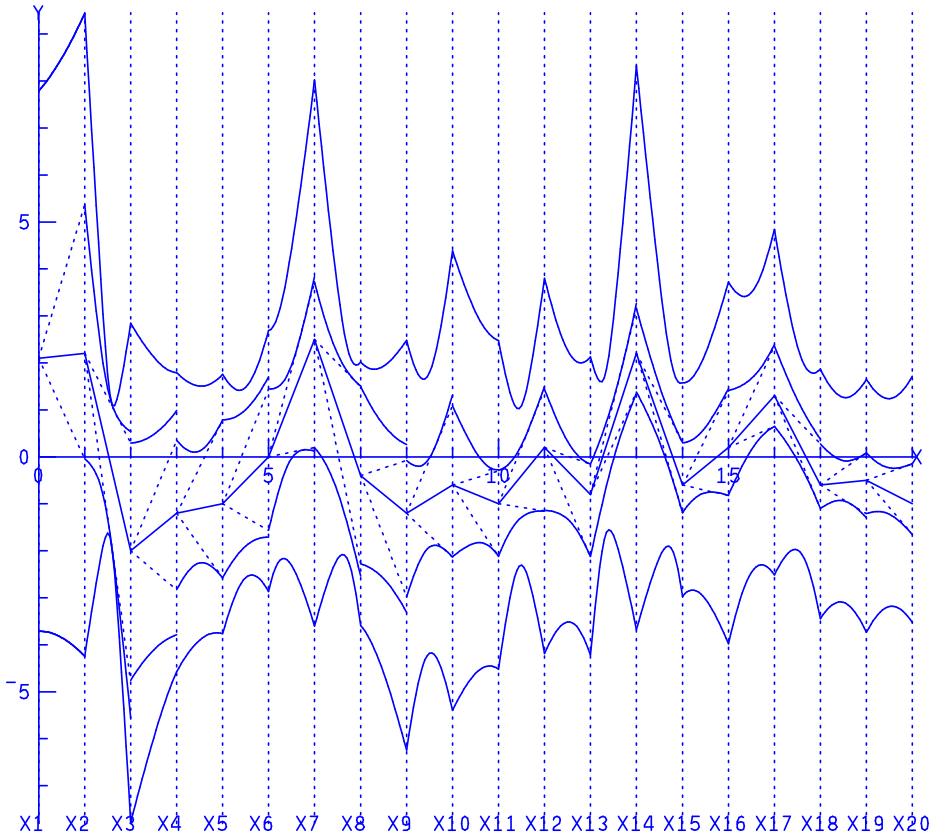

Figure 92: Finding a Feasible Point – state of the system – for a Process Represented by the Hypersurface. A process being a relation among several variables can be represented by a hypersurface. A feasible state of the system involved corresponds to an interior point of the hypersurface – since all the constraints are satisfied simultaneously. The intermediate envelopes on both sides of the polygonal line indicate the local curvature of the hypersurface in a neighborhood of the point. Notice that here X_{13} , X_{14} , X_{15} are the *critical variables* since the available ranges involved – for maintaining control – are the narrowest. The display shown can serve as the systems intrumentation. As a value of a variable is fixed the display provides the *available range* for the remaining variables.

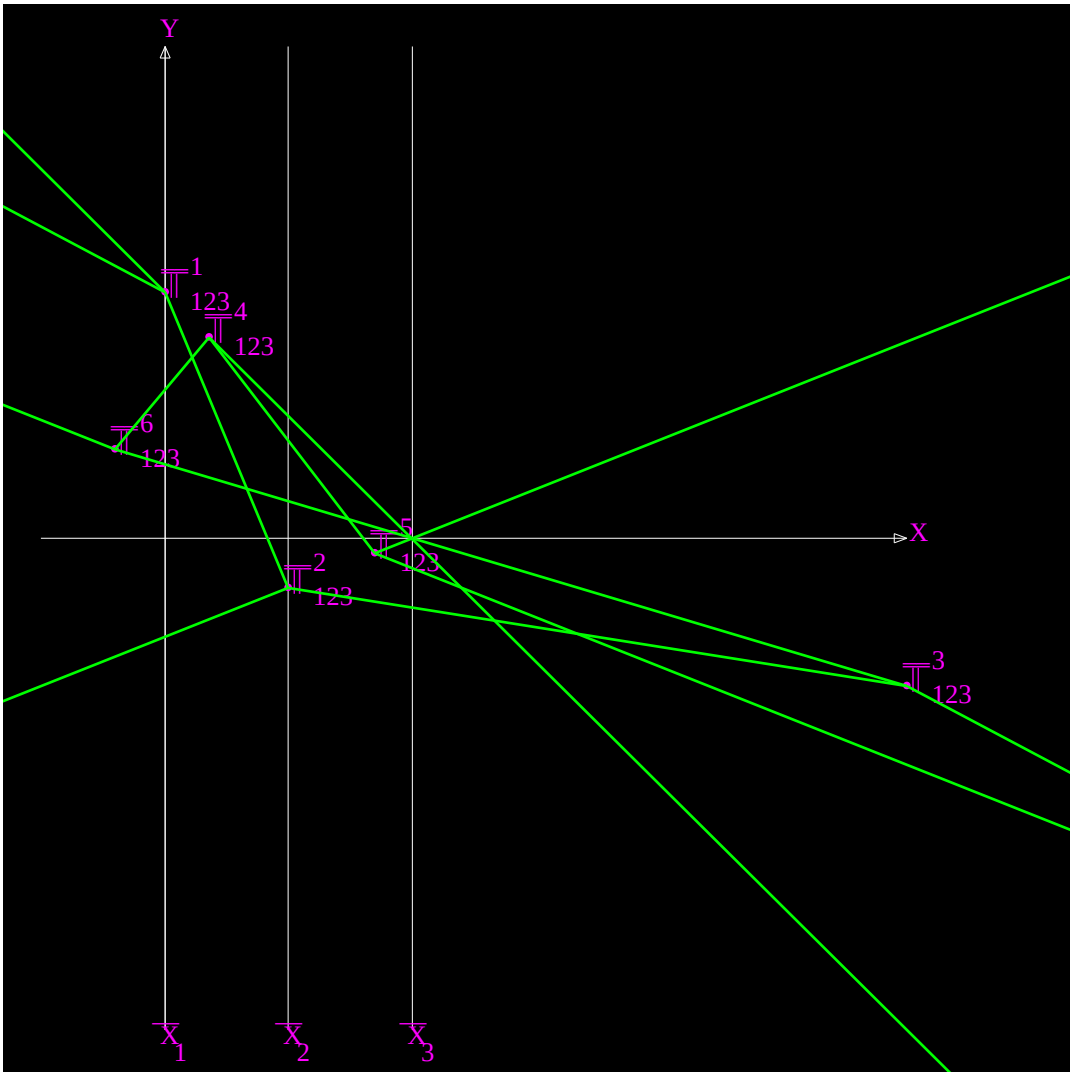

Figure 93: Adjacency relationship of the 2-faces of the convex 3-polytope in Parallel Coordinates

The 123 representation of the 6 2-flats containing the 6 2-faces of the 3-polytope is shown. The 3-polytope in this example being convex, all the adjacency relationships are represented by line segments.

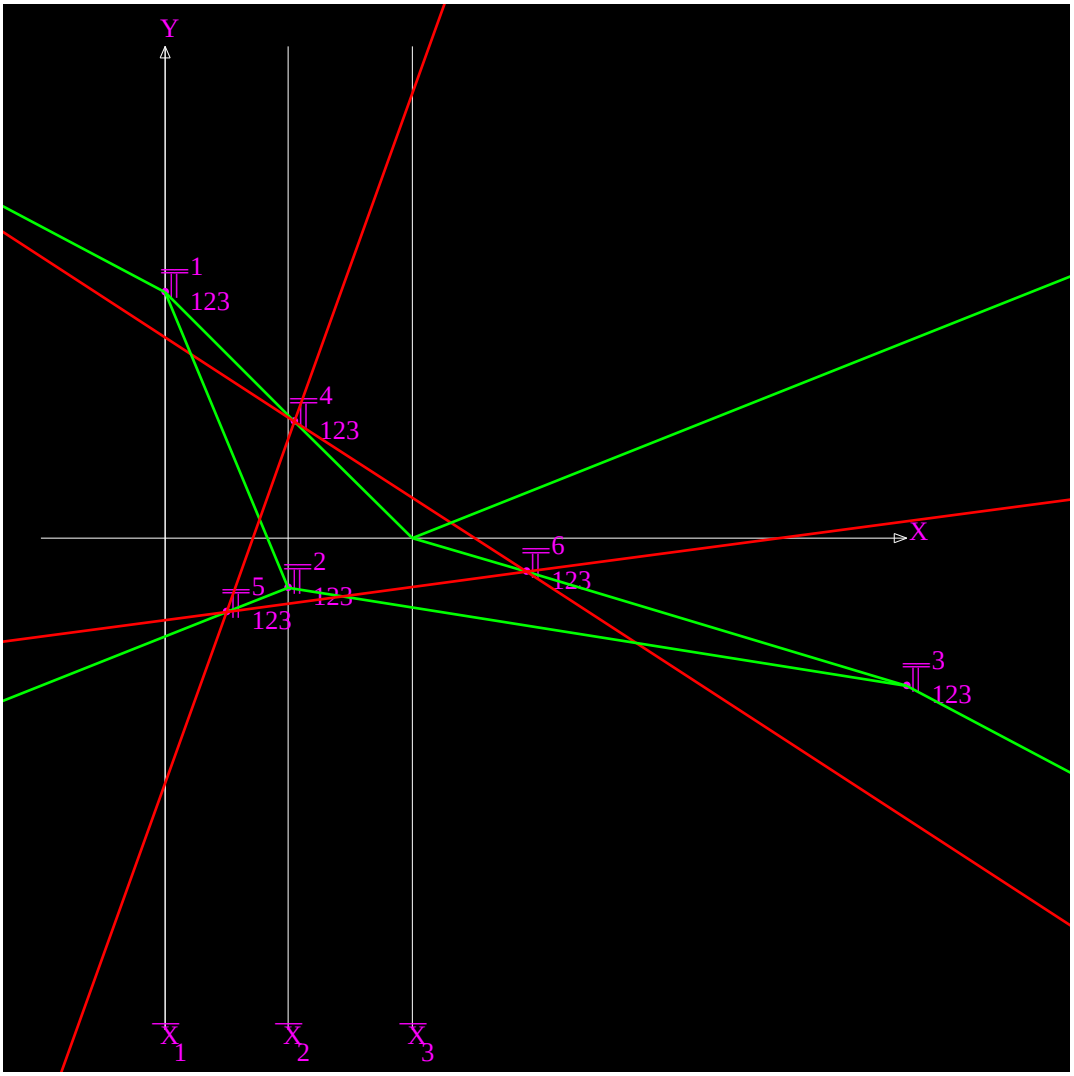

Figure 94: Adjacency relationship of the 2-faces of the non-convex 3-polytope in Parallel Coordinates

The 123 representation of the 6 2-flats containing the 6 2-faces of the 3-polytope is shown. The 3-polytope in this example being non-convex has some adjacency relationships ($\overline{\pi}^4_{123}$ and $\overline{\pi}^5_{123}$, $\overline{\pi}^5_{123}$ and $\overline{\pi}^6_{123}$, $\overline{\pi}^6_{123}$ and $\overline{\pi}^4_{123}$) which are represented by lines instead of line-segments.

Representing surfaces by their tangent planes – see also next section.

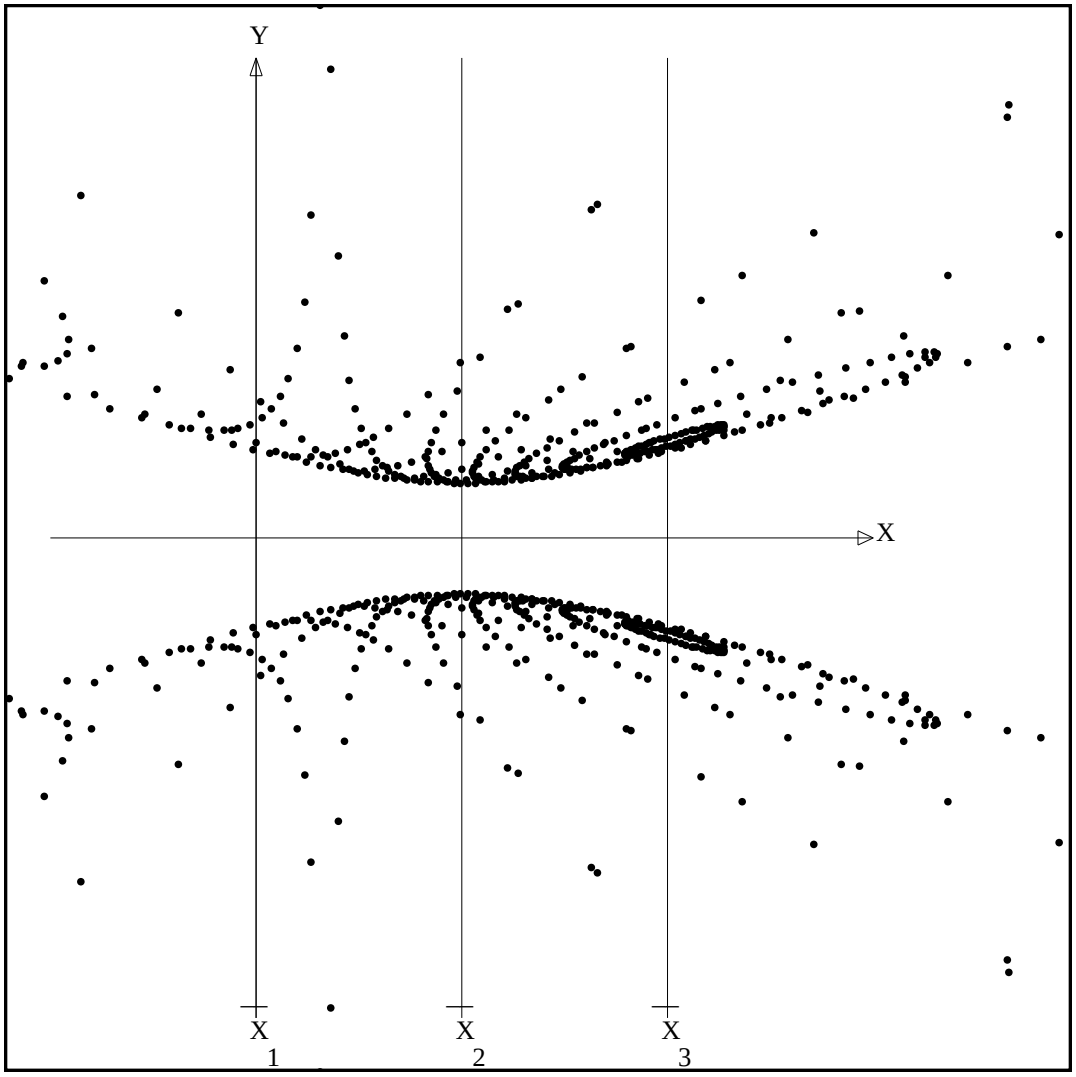

Figure 95: A Sphere in 3-D represented by its tangent planes (points). The hyperbolic pattern of the envelopes indicates that the object is **convex**.

The conjecture is that with the tangent plane representation convex objects in N-D are represented by generalized hyperbolas – see gconics.

REPRESENTING SURFACES IN TERMS OF THEIR TANGENT PLANES

Chao-Kuei Hung @ USC

DEVELOPABLE SURFACES – QUADRICS

CONICS $\rightarrow$ CONICS

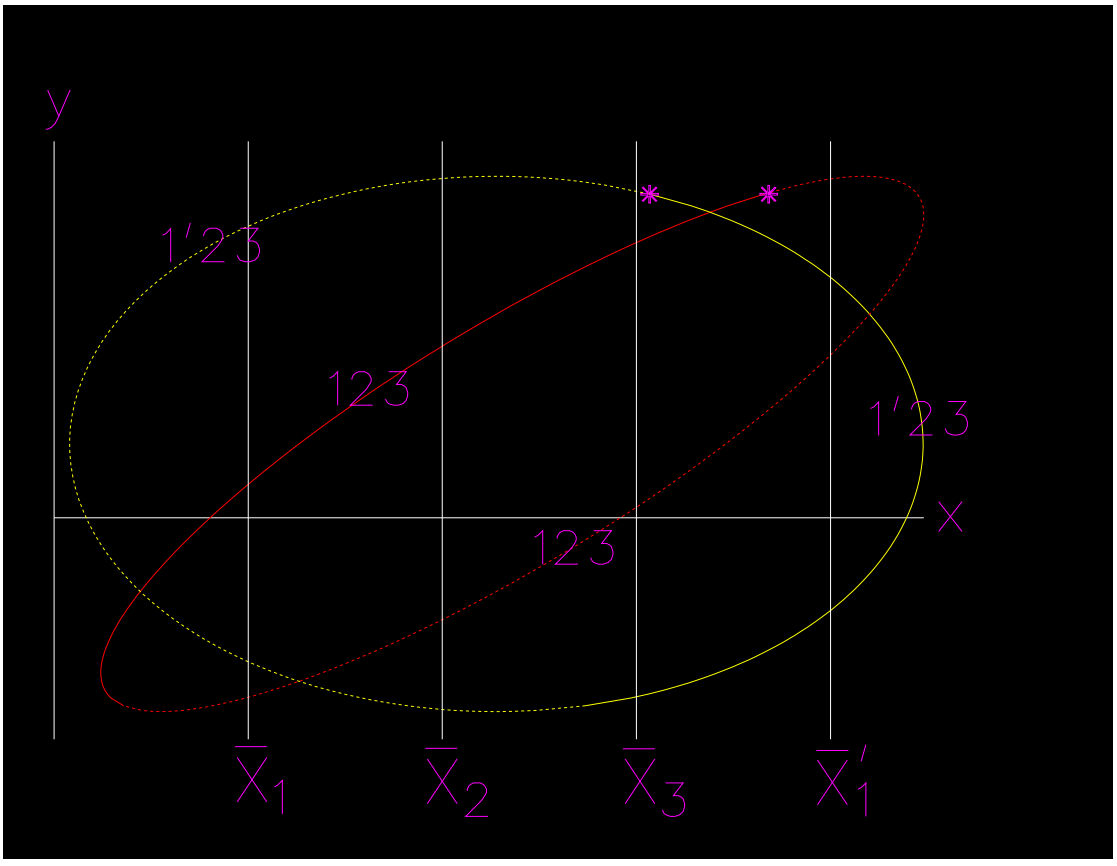

Figure 96: Representation is a pair of ellipses

Cone vertex is (0,0,1), axis vector is (6,8,7), circle center is at (6,8,8), radius is 5.

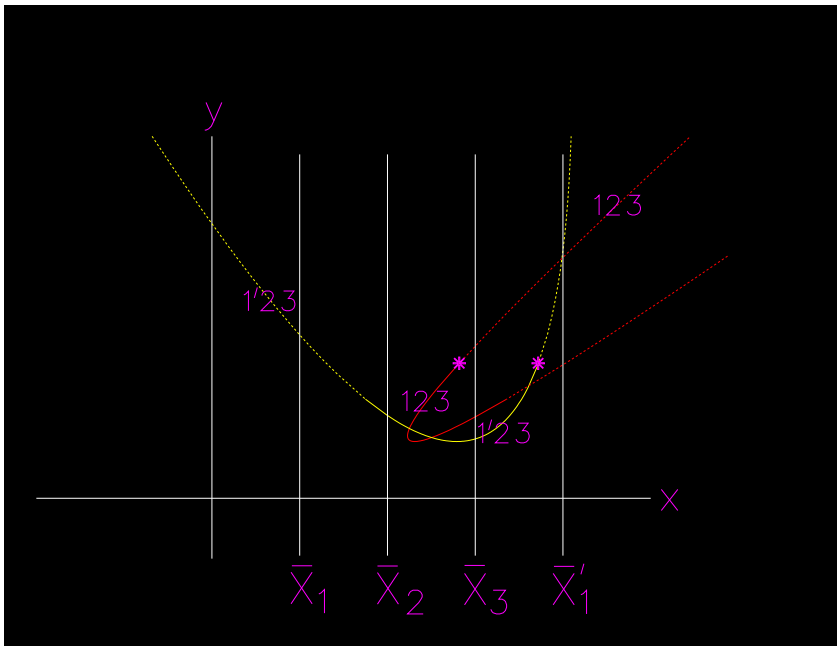

Figure 97: Representation is a pair of parabolas

Cone vertex is (0,0,1), axis vector is (-0.6,0.8,5), circle center is at (-0.6,0.8,6), radius is 7.

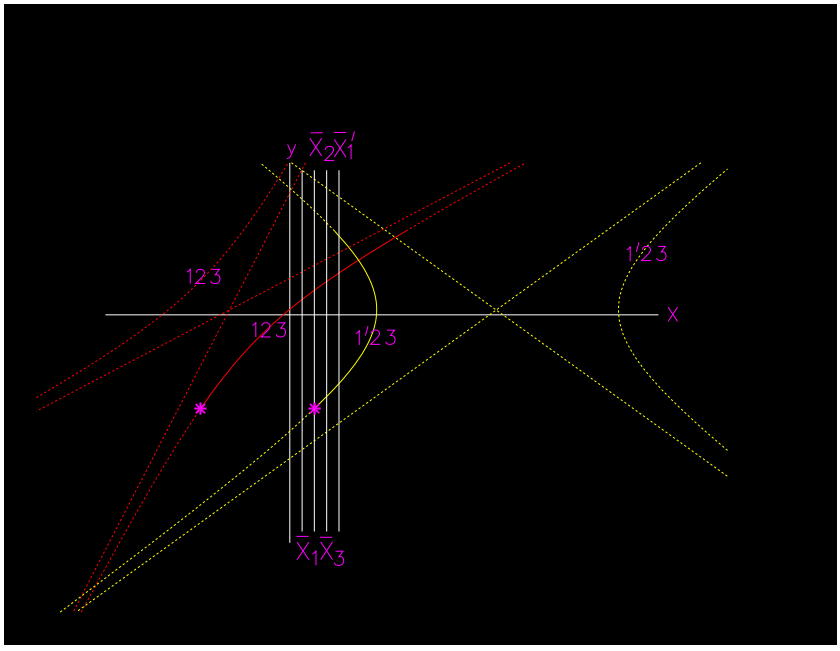

Figure 98: Representation is a pair of hyperbolas

Cone vertex is at (0,0,1), axis vector is (6,8,7), circle center is at (6,8,8), radius is 1.

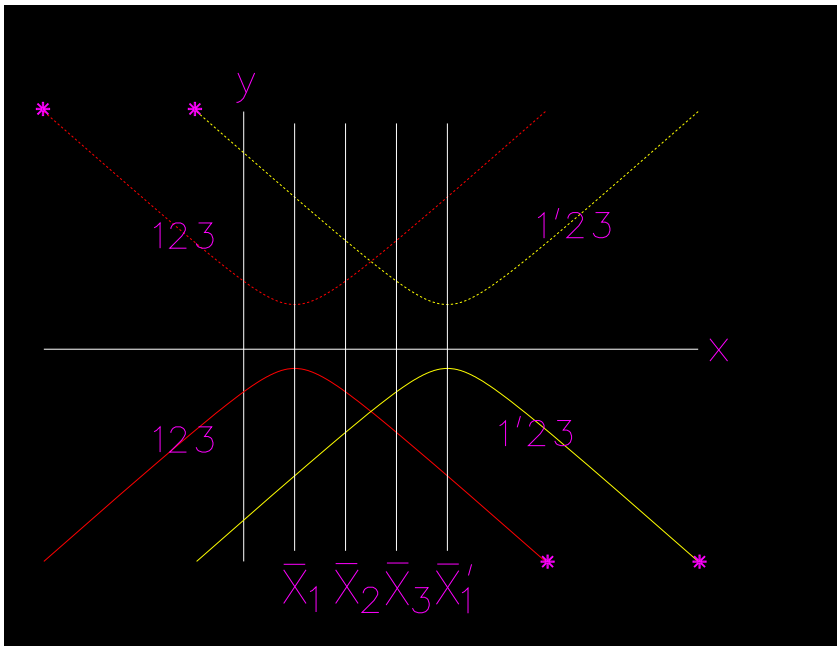

Figure 99: Representation is a pair of hyperbolas

Representation of cylinder with axis defined by the points (2,2,2), (2,3,3), radius is 5.

Ruled Surfaces

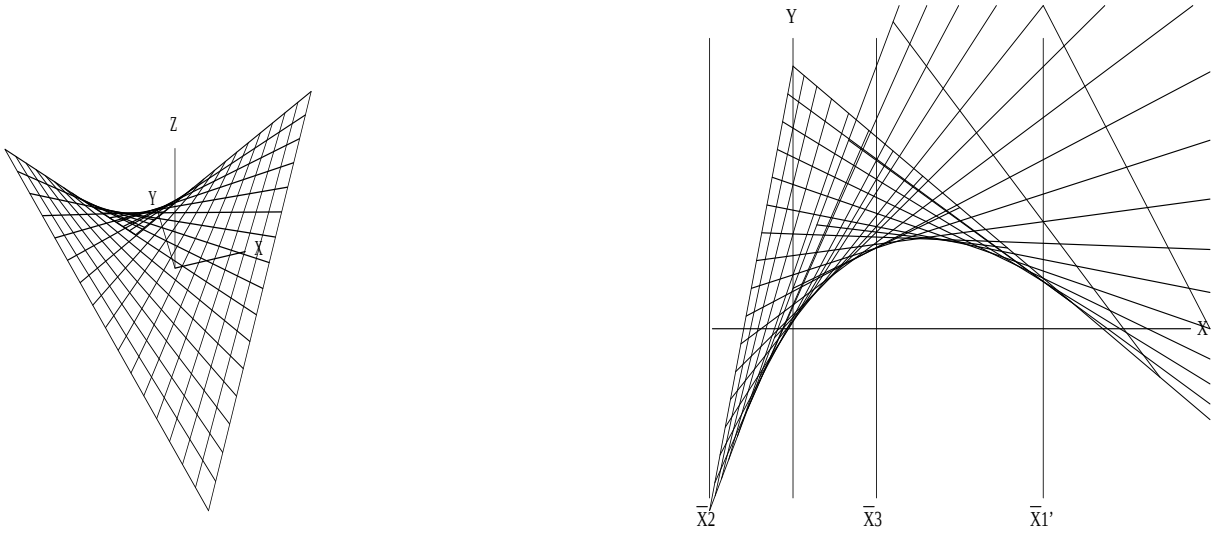

Figure 100: Hyperbolic paraboloid - Sampling along rulings gives meshes of straight lines – self-dual.

Finally we illustrate the methodology's ability to model multivariate relations in terms of hypersurfaces – just as we model a relation between two variables by a planar region. Then by using the interior point algorithm, with the model we can do trade-off analyses, discover sensitivities, understand the impact of constraints, and in some cases do optimization. For this purpose we shall use a dataset consisting of the outputs of various economic sectors and other expenditures of a particular (and real) country. It consists of the monetary values over several years for the Agricultural, Fishing, and Mining sector outputs, Manufacturing and Construction industries, together with Government, Miscellaneous spending and resulting GNP; eight variables altogether. We will not take up the full ramifications of constructing a model from data. Rather, we want to illustrate how ||-coords may be used as a modeling tool. Using the Least Squares technique we "fit" a function to this dataset and we are not concerned at this stage whether the choice of function is a "good" choice or not. The function we obtained bounds a region in R^8 and is represented by the upper and lower curves shown in Fig. 101.

The picture is in effect a simplistic *visual* model of the country's economy, incorporating it's capabilities, limitations and interelationships among the sectors etc. A

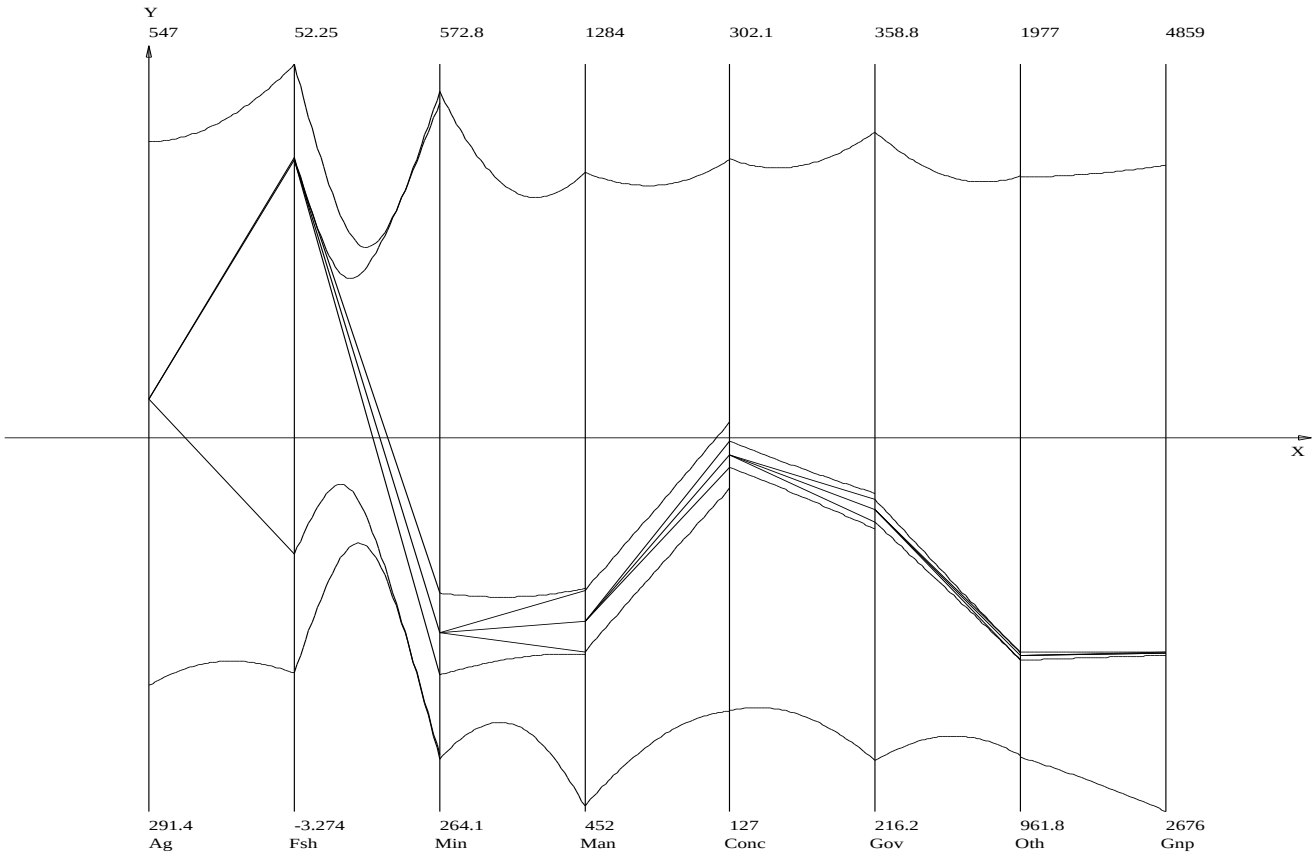

Figure 101: Model of a country's economy

point interior to the region, satisfies all the constraints simultaneously, and therefore represents (i.e. the 8-tuple of values) a *feasible economic policy* for that country. Using the interior point algorithm we can construct such points. It can be done interactively by sequentially choosing values of the variables and we see the result of one such choice in Fig. 101. Once a value of the first variable is chosen (in this case the agricultural output) within it's range, the dimensionality of the region is

reduced by one. In fact, the upper and lower curves between the 2nd and 3rd axes correspond to the resulting 7-dimensional hypersurface and show the *available* range of the second variable (Fishing) reduced by the constraint. In fact, this can be seen (but not shown here) for the rest of the variables. That is, due to the relationship between the 8 variables, a constraint on one of them impacts all the remaining ones and restricts their range. The display allows us to experiment and actually see the impact of such decisions "downstream". By interactively varying the chosen value for the first variable we found, that it not possible to have a policy that favors Agriculture without also favoring Fishing and vice versa.

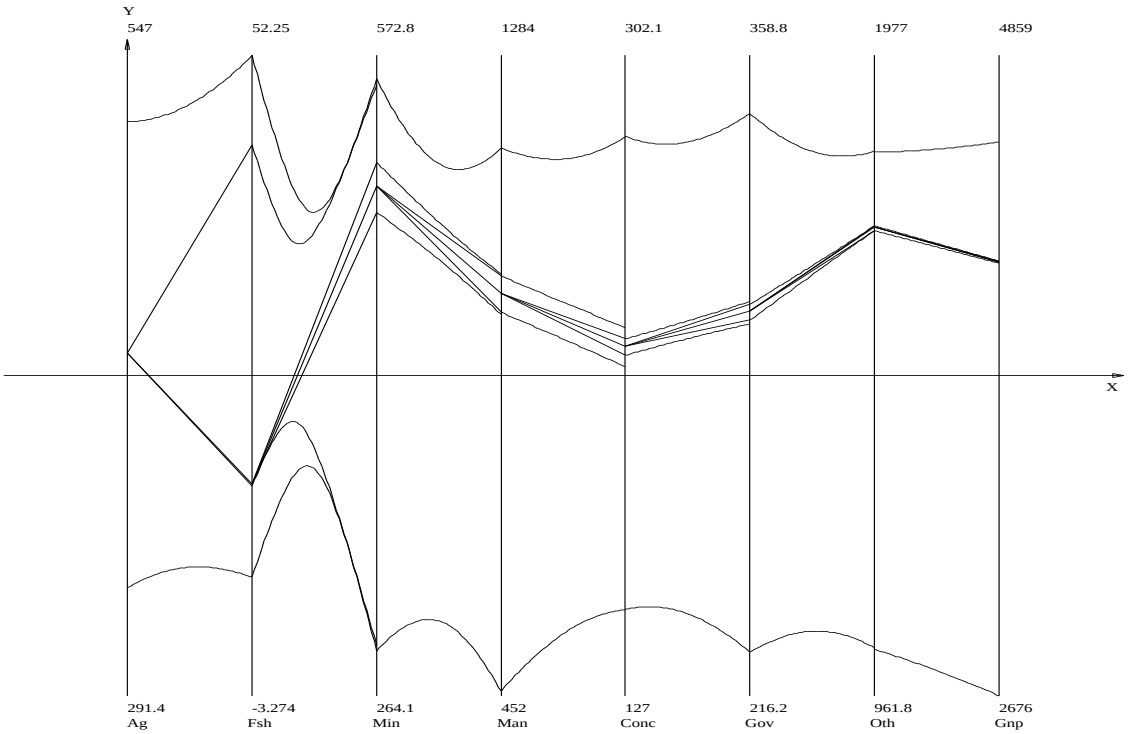

Figure 102: Competition for labor between the Fishing & Mining sectors – compare with previous figure

Proceeding, a very high value from the available range of Fishing is chosen and it corresponds to very low values of the Mining sector. By contrast in Fig. 101 we see that a low value in Fishing yields high values for the Mining sector. This inverse correlation was examined and it was found that the country in question has a large number of *migrating* semi-skilled workers. When the fishing industry is doing well most of them are attracted to it leaving few available to work in the mines and vice versa. The comparison between the two figures shows the *competition for the same resource* between Mining and Fishing. It is especially instructive to discover this interactively. The construction of the interior point proceeds in the same way.

A theorem guarantees that a polygonal line which is in-between all the intermediate curves/envelopes represents an interior point of the hypersurface and all interior points can be found in this way. If the polygonal line is tangent to anyone of the intermediate curves then it represents a *boundary point,* while if it crosses anyone of the intermediate curves it represents an *exterior point.* The later enables us to see, in an application, the first variable for which the construction failed and what is needed to make corrections. By varying the choice of value over the available range of the variable interactively, sensitive regions (where small changes produce large changes downstream) and other properties of the model can be easily discovered.

Once the construction of a point is completed it is possible to vary the values of each variable and see how this effects the remaining variables. So one can do *trade-off analysis* in this way and provide a powerful tool for, Decision Support, Process Control and other applications. As new data becomes available the model can be updated with the Decision Making being based on the most recent information.

APPENDIX A – BIBLIOGRAPHY ON MULTIVARIATE MULTI-DIMENSIONAL VISUALIZATION

Bibliography

[1] **C. Ahlberg and B. Shneiderman (1994), Visual Information Seeking : Tight Coupling of Dynamic Query Filters with Starfield Displays CHI'94, ACM, Boston, 313-317**

[2] **B. Alpern, L. Carter (1991), The Hyperbox, IEEE Conf. Visualization '91, 133-39**

[3] **D. F. Andrews (1972), Plots of High-Dimensional Data, Biometrics, 29, 125-136**

[4] **R. Arnheim (1969), Visual Thinking, Univ. of Cal. Press, Berkeley**

[5] **D. Asimov (1985), The Grand Tour: A Tool For Viewing Multidimensional Data, SIAM J. of Sci. & Stat. Comp., 6, 128-143**

[6] **T. F. Banchoff (1990), Beyond the Third Dimension, Scientific American Library, NY**

[7] **V. Barnett Edit. (1981), Interpreting Multivariate Data, Wiley, New York**

[8] **R. A. Becker, W. S. Cleveland , A. R. Wilks (1988), Dynamic Graphics for Statistics, Wadsworth, Belmont, CA**

[9] **R.A.Becker, S. G. Eick, and A. R. Wilks (1995), Visualizing Network Data, IEEE Transactions on Visualization and Computer Graphics, 1, 16-28**

[10] **J. Beddow (1990), Shape Coding of Multidimensional Data on a Microcomputer display, Proc. of IEEE Conf. Visualization ’90, 238-46**

[11] **R. D. Bergeron, G. Grinstein (1989), A Reference Model For The Visualization of Multi-Dimensional Data, Proc. of Eurographics ’89**

[12] **J. Bertin (1967), Semiologie Graphique, Gauthier-Villars, Paris**

[13] **D. Brissom Edit. (1979), Hypergraphics: Visualizing Complex Relationships in Art, Science and Technology, Amer. Assoc. for the Adv. of Sc., Westview Press, Boulder**

[14] **A. Buja, C. Hurley, J. A. McDonald (1986), A Data Viewer for Multivariate Data, in Proc. 18th Interface Symp., 171-4**

[15] **J. M. Chambers, W. S. Cleveland, B. Kleiner , P. Tukey (1983) Graphical Methods for Data Analysis, Wadsworth, Belmont, CA**

[16] **H. Chernoff (1973), The Use of Faces to Represent Points in k-Dimensional Space Graphically, J. Amer. Stat. Assoc., 68, 361-368**

[17] **E. Cluff, R. P. Burton, W. A. Barrett (1991), A Survey and Characterization of Multidimensional Presentation Techniques, J. Imag. Tech., 17-4, 142-153**

[18] **D. Cox (1988), Using the Supercomputer to Visualize Higher-Dimensions: An Artist's Contribution to Scientific Visualization, Leonardo: Journal of Art, Science and Tech., 21-3, 233-242**

[19] **S. L. Crawford , T. C. Fall (1990), Projection Pursuit Techniques for the Visualization of High Dimensional Datasets, in Visualization in Scientific Computing, G. M. Nielson & B. Shriver Ed., IEEE Computer Soc. Press, Los Alamitos, CA, 94-108**

[20] **T. A. Defanti , M. D. Brown (1989), Insight Trough Images, Unix Rev., 7-3, 42-50**

[21] **T. A. Defanti, M. D. Brown , B. H. McCormick (1989), Visualization Expanding Scientific and Engineering Research Opportunities, Computer, 22-8, 12-25**

[22] **P. Diaconis , J. H. Friedman (1980), M and N plots, Stanford Lin. Acc. Center Rep., PUB-2495**

[23] **S. G. Eick (2000), Visual Discovery and Analysis, IEEE Transactions on Visualization and Computer Graphics, 6, 44-58**

[24] **K. H. Esbensen (2000) Multivariate Data Analysis - in practise. An introduction to multivariate data analysis and experimental, 5th ed., CAMO Process AS Publ. Oslo, Norway**

[25] **U. Fayyad, G. Grinstein and A. Wierse (2001), Information Visualization in Data Mining and Knowledge Discovery, Morgan-Kaufmann Publishers,**

[26] **S. Feiner , C. Beshers (1990), Visualizing n-Dimensional Virtual Worlds with n-Vision, Computer Graphics, 24-2, 37-8**

[27] **M. Friendly (1999) Visualizing Categorical Data, Cognition and Survey Research, M. Sirken et al. editors, John Wiley, New York 319-348**

[28] **J. Friedman , J. Tukey (1974), A Projection Pursuit Algorithm for Exploratory Data Analysis, IEEE Trans. Comp., C-23, 881-890**

[29] **G.W. Furnas, A. Buja (1994) Prosection Views: Dimensional Inference through Sections and Projections, J. Comp. and Graph. Stat., 3, 323-385**

[30] **G. Grinstein , S. Smith (1990), The Perceptualization of Scientific Data, in Proc. SPIE (Int. Soc. Opt. Eng.) Extracting Meaning from Complex Data: Processing, Display, Interaction, Santa Clara, CA, 1259, 190-9**

[31] **J.L. Helman , L. Hesselink (1991), Visualizing Vector Field Topology in Fluid Flows, IEEE Comp. Graph. and appl., 11-3, 36-46**

[32] **P. J. Huber (1985), Projection Pursuit, Ann. Stat., 13, 435-475**

[33] **C. Hurley and A. Buja (1990), Analyzing High-Dimensional Data with Motion Graphics, SIAM J. Sci. Stat. Comput., 11-6, 1193-1211**

[34] **A. Inselberg (1985), The Plane with Parallel Coordinates, Special Issue on Computational Geometry of The Visual Computer, 1, 69-97**

[35] **A. Inselberg, B. Dimsdale (1990), Parallel Coordinates : A Tool for Visualizing Multi-Dimensional Geometry, Proc. VISUALIZATION 90, IEEE Computer Society Press, Los Alamitos, CA , 361-378**

[36] **D. A. Keim (2000), Designing Pixel-Oriented Visualization Techniques: Theory and Applications, IEEE Transactions on Visualization and Computer Graphics, 6, 59-78**

[37] **B. Kleiner, J. A. Hartigan (1981), Representing Points in Many Dimensions by Trees and Castles , J. Amer. Stat. Assoc., 76, 260-269**

[38] **O. Lathrop (1989) State of the Art in Data Visualization, ACM SIGGRAPH ’89 Course Notes, Course Number 28, Boston, Mass. 1989**

[39] **J. LeBlanc, M. O. Ward, N. Wittels (1990), Exploring N-Dimensional Databases, Proc. of IEEE Conf. VISUALIZATION ’90, 230-7,**

[40] **U. Cvek, A. Gee, P. Hoffman, D. Pinkney, M. Trutschl, H. Zhang, K. Marx, and G. Grinstein (1999), Data Mining of Yeast Functional Genomics Data Using Multidimensional Analytic and Visualization Techniques, Drug Discovery Technology 1999, Boston,**

[41] **B. H. McCormick, T. A. Defanti , M. D. Brown (1987), Visualization in Scientific Computing, Computer Graphics, 21, 6, ACM SIGGRAPH: New York,**

[42] **T. Mihalisin, J. Timlin, J. Schwegler (1991), Visualizing Multivariate Functions, Data and Distributions, IEEE Comp. Graph. and appl., 11-3, 28-35**

[43] **G.M. Nelson, T. Foley, B. Hamann , D. Lane (1991), Visualizing and Modeling Scattered Multivariate Data, IEEE Comp. Graph. and appl., 11-3, 47-55**

[44] **R.M. Pickett , H. Levkowitz (1990), Iconographic Displays of Multiparameter and Multimodal Images, 1st Conf. on Vis. in Biomed. Imag, Atlanta**

[45] **P.K. Robertson (1991), A Methodology for Choosing Data Representations, IEEE Comp. Graph. and appl., 11-3, 56-67**

[46] **B. Schneiderman (1996), The Eyes Have It – a Task by Data Type Taxonomy for Information Visualization Proceedings of IEEE/CS Symposium Visual Languages (VL'96), 336-343**

[47] **D. W. Scott (1986), Data Analysis in 3 and 4 Dimensions With Nonparametric Density Estimation, in [55], 291-305**

[48] **A. J. Smith, J. Nelder, A. Buja, Z. Malik, L. Tweedie, R. Spence. Sampling Schemes for Model Visualisation to appear: J. Comp. and Grap. Stat.**

[49] **L.S. Tierney (1990), LISP-STAT: An Object-Oriented Environment for Statistical Computing and Dynamic Graphics, Wiley, New York**

[50] **E. R. Tufte (1983), The Visual Display of Quantitative Information, Graphic Press, Cheshire, Conn.**

[51] **E. R. Tufte (1990), Envisioning Information Graphic Press, Cheshire, Conn.**

[52] **E.R. Tufte (1996), Visual Explanation, Graphic Press, Cheshire, Conn.**

[53] **L. Tweedie, B. Spence, D. Williams, and R. Bhogal, R. (1994), The Attribute Explorer CHI'94, Boston,**

[54] **A. R.Unwin, G. Hawkins, H. Hofmann, and B. Siegl, (1996). Interactive Graphics for Data Sets with Missing Values - MANET, J. of Comp. and Grap. Stat., 5(2), 113-122.**

[55] **A. R. Unwin, (1999). Requirements for interactive graphics software for exploratory data analysis, Comp. Stat., 14, 7-22.**

[56] **G. J. Wills (1996), Selection: 524,288 Ways to Say "This Is Interesting",Proceedings of IEEE Symposium on Information Visualization (InfoVis '96), 54-60**

[57] **P.C. Wong and D.R. Bergeron (1997), 30 years of Multidimensional Multivariate Visualization in Scientific Visualization Overviews Methodologies & Techniques, IEEE Computer Society Press**

APPENDIX B – BIBLIOGRAPHY ON PARALLEL COORDINATES

Bibliography

[1] **G. Andrienko and N. Adrienko Constructing Parallel Coordinates Plot for Problem Solving, 1st International Symposium on Smart Graphics, Hawthorne, New York,**

[2] **T. Avidan, and S. Avidan (1998) Parallax - A data mining tool based on parallel coordinates, Comp. Stat. 13-1 65–**

[3] **O.M.Becker (1997), Representing Protein and Peptide Structures with Parallel-Coordinates, J. Comput. Chem. 18, 1893-1902.**

[4] **C. Behrens (1990), Multivariate Procedures for the Cooperative Analysis of ”Texture” in Time Allocation Data, Presented at the Society for Cross-Cultural Research Symposium, Claremont, CA, 1990 Proc., 1-37**

[5] **M.R. Berthold and L.O. Hall (1999), Visualizing Fuzzy Points in Parallel Coordinates, Tech. Rep. UCB/CSB -99-1082, Univ. of Cal. Berkeley**

[6] **A. Chatterjee (1995), Visualizing Multidimensional Polytopes and Topologies for Tolerances, Ph.D. Thesis, Univ. of S. Calif. Dept. of Computer Science,**

[7] **A. Chatterjee, P. Das, and S. Bhattacharya, (1993), Visualization in Linear Programming using Parallel Coordinates, J. Pattern Rec. 26-11, 1725–36**

[8] **T. Chomut (1987), Exploratory Data Analysis Using Parallel Coordinates, M. Sc. Thesis, UCLA Dept. of Computer Science, IBM Los Angeles Scientific Center Report, 1987-2811**

[9] **S. M. Cohan, D. C. H. Yang (1986), Mobility Analysis of Planar Four-Bar Mechanisms through the Parallel Coordinate System, J. of Mech. & Mach. Th., 21, 63-71**

[10] **D. Cook (1990), Comments on Unmasking Multivariate Outlier Leverage Points, J. Amer. Stat. Assoc, 85 , 633**

[11] **D. B. Curtiss, R. P. Burton (1991), Comparison of Parallel Axes Graphics and Binocular Parallax Graphics, J. Imag. Tech., 17-4, 154-156**

[12] **A. Desai, L. C. Walters (1991), Graphical Presentations of Data Envelopment Analyses: Management Implications from Parallel Axes Representations, J. Decision Sc., 22-2, 335-353**

[13] **J. A. Dykes (1997), Exploring spatial data representation with dynamic graphics, Computers and Geosciences, 23(4), 345-370**

[14] **R. Finsterwalder (1991), A Parallel Coordinate Editor as a Visual Decision Aid in Multi-Objective Concurrent Control Engineering Environment IFAC CAD Contr. Sys., Swansea, UK 119-122**

[15] **P. Fiorini, A. Inselberg (1989), Configuration Space Representation in Parallel Coordinates in Proc. of 1989 IEEE Inter. Conf. on Robotics and Automation, IEEE Computer Society, 1215-1220**

[16] **C. Gennings, K.S. Dawson, W.H Carter, R.H. Myers (1990), Interpreting Plots of a Multidimensional Dose-Response Surface in a Parallel Coordinate System Biometrics, 46, 719-735**

[17] **J. J. Helly (1987), Application of Parallel Coordinates to Complex System Design and Operation, in Proc. of NCGA 1987, Fairfax, VA, vol III, 541-546**

[18] **H. Hinterberger (1987), Data Density : A Powerful Abstraction to Manage and Analyze Multivariate Data, Ph. D. Thesis, ETH , #4 VDF, Zurich**

[19] **A. Inselberg (1985), The Plane with Parallel Coordinates, Special Issue on Computational Geometry of The Visual Computer, 1, 69-97**

[20] **A. Inselberg (1985), Intelligent Instrumentation & Process Control, in Proc. IEEE Confer. on Artificial Intelligence Applications, Miami, 302-307**

[21] **A. Inselberg (1989), Discovering Multi-Dimensional Structure using Parallel Coordinates, (Invited Paper), Proc. Amer. Stat. Assoc. 150th Ann Conf., Sec. on Stat. Graphics, 1-16**

[22] **A. Inselberg (1990), Parallel Coordinates: A Tool For Visualizing Multi-Dimensional Geometry, in IEEE Comp. Soc. Proc. of Visualization '90 Conf, San Francisco, 361-378**

[23] **A. Inselberg, M. Boz, B. Dimsdale (1991), Conflict Resolution, One-Shot Problem and Air Traffic Control, sub. for pub., available as IBM Palo Alto Sci. Cent. Tech. Rep., G320-3559**

[24] **A. Inselberg, B. Dimsdale (1994), Multi-Dimensional Lines I: Representation, SIAM J. Appl. Math., Vol 54, No. 2, pp. 559-577.**

[25] **A. Inselberg, B. Dimsdale (1994), Multi-Dimensional Lines II : Proximity & Applications, SIAM J. Appl. Math., Vol 54, No. 2, pp. 578-596.**

[26] **A. Inselberg, B. Dimsdale, J. Eickmeyer, C. Genning (1991), Parallel Coordinates for Multi-Dimensional Geometry: Foundations and Applications, Invited Special Session at ICIAM 91 (Inter. Congress on Ind. & Applied Math, in Proc.)**

[27] **A. Inselberg, C. K. Hung (1991), A New Representation of Hypersurfaces in Terms of Tangent Hyperplanes, Proc. of SIAM Conference on Geometric Design, 11**

[28] **A. Inselberg, M. Reif , T. Chomut (1987), Convexity Algorithms in Parallel Coordinates, J. of ACM, 34, 765-801**

[29] **A. Inselberg (1998), Visual Data Mining with Parallel Coordinates, Comp. Stat. 13-1 47–64**

[30] **A. Inselberg (1999), Don’t Panic ... Do it in Parallel, Comp. Stat. 14 53–77**

[31] **A. Inselberg and T. Avidan (1999) The Automated Multidimensional Detective, Proc. of IEEE Infor. Vis. '99, 112-119, IEEE Comp. Soc., Los Alamitos, CA**

[32] **Z. Izhakian (2001), An Algorithm for Computing A Polynomial's Dual Curve in Parallel Coordinates, M.Sc. Thesis, Comp. Science Dept. Tel-Aviv Univ.**

[33] **C. V. Jones (1996) Visualization and Optimization, Kluwer Academic Publishers, Boston**

[34] **D. A. Keim and H. P. Kriegel, Visualization Techniques for Mining Large Databases: A Comparison, TKDE, 8-6, 923–938,**

[35] **B.T. Luke (1992), Theoretical Examinations of Polypeptide Folding, sub. for pub. to J. of Amer. Chem. Soc.**

[36] **M.O. Ward (1994), XmdvTool: integrating multiple methods for visualizing multivariate data, Proc. IEEE Conf. on Visualization, San Jose, CA, 326-333**

[37] **A. R. Martin and M. O. Ward (1995), High dimensional brushing for interactive exploration of multivariate data, Proc. IEEE Conf. on Visualization, Atlanta, GA, 271-278**

[38] **J. Rivero, A. Inselberg (1984), Extension al analisis del Espacio De Fase de Systemas Dinamicos por las Coordenadas Paralelas, IBM LA Sci. Cen. Rep., G320-2742, Presented at VII Systems Engr. Workshop, Santiago Chile**

[39] **R. Wegenkittl, H. Loffelmann, and E. Groller (1997), Visualizing the behaviour of higher dimensional dynamical systems, Prof. of IEEE Vis. Conf.**

[40] **E. Wegman (1990), Hyperdimensional Data Analysis Using Parallel Coordinates, J. Amer. Stat. Assoc., 85, 664-675**

wwwsites

- **Milestones in the history of thematic cartography, statistical graphics, and data visualization – M.Friendly and D. J. Denis 2001 – http://www.math.yorku.ca/SCS/Gallery/milestone/**

- **Visualization of domain and concept descriptions – www-ai.ijs.si/DunjaMladenic/papers/abstract/ascai92.html**
- **Directions in Spatial Spectroscopy – www.math.yorku.ca/Who/Faculty/Monette/pub/s-99a**
- **Welcome to Starlight - Remote Sensing Group Project – www.pnl.gov/remote/projects/starlight/theory.html**
- **Breaking the Barriers of 3D Visualization – www.sv.vt.edu/future/muri/white/white.html**
- **Extruded Parallel Coordinates – www.cg.tuwien.ac.at/ rubik/extruded.htm**
- **Links to my Master's Thesis – www.ifs.tuwien.ac.at/ rkosara/thesislinks.html (In this site there a particularly well organized literature review)**
- **Hierarchical Parallel Coordinates – avis.wpi.edu/ matt/courses/parcoord**
- **Parallel Coordinates for Power Stability – www.caip.rutgers.edu/ peskin/epriRpt/PowerStability.html**

- **Thermodynamic Cycle Data – www.caip.rutgers.edu/ peskin/epriRpt/ThermoCycle.html**
- **Parallel Coordinates in MATHLAB – www.math.tau.ac.il/ nin/learn98/Nauman/parplot.html**
- **The PARCOVI Project-The Parallel Coordinates Visualizer – atkosoft.com/statparcovi.htm**
- **Visualization of a THERMOPOT – www.inf.ethz.ch/personal/lindenme/thermoprot**
- **DNA VISUAL AND ANALYTIC DATA MINING – www.cs.uml.edu/ phoffman/dna1**
- **Kohonen neural network visualizations – www.anvilinformatics.com/portfolio/yeast/yeaste.html**
- **A Visual Approach for Monitoring Logs – www.usenix.org/publications/library/proceedings/lisa98**
- **Visualizing Large Datasets – www1.math.uni-augsburg.de/ unwin/AntonyArts/visualising.html**
- **Visualisation Techniques for Statistics – www1.math.uni-augsburg.de/ unwin/AntonyArts/VisTechNTTS.html (Excellent**

site)

- **Exploratory Data Visualiser – www1.bell-labs.com/user/gwills/EDVguide/guide/guide.html**
- **Nonlinear Feature Space Transformations – www.cs.unc.edu/ cog-gins/Research/Nonlinear/NonlinearPaper.html**
- **AN INVESTIGATION OF FUNDAMENTAL FREQUENCIES OF LAMINATED CIRCULAR CYLINDERS– www.knowledgestor.com**
- **Evaluation of Marine Data by visual means – www.egd.igd.fhg.de**
- **Manual Endmember Selection Tool – cires.colorado.edu/cses/research**
- **Vizcraft - Multidimensional Visualization of Aircraft Design – cs-grad.cs.vt.edu/ agoel/vizcraft.html**
- **Parallel Coordinates Visualization Applet – cs-grad.cs.vt.edu/ agoel/parallel-coordinates**
- **Public Policy Analysis –www.ppm.ohio-state.edu/ppm/research-groups/pubpolan.html**
- **ACM Digital Library Reconnaissance sup-**

port for juggling multiple processing – www.acm.org/pubs/citations/proceedings/uist/192426/p27-lunzer

- **Information Visualization: Data Types – www.cs.umd.edu/hcil/pubs/presentations/eyeshaveit/tsld013.htm (This is an important site for InfoVis)**