

# Theory of Dynamic Projections in High-Dimensional Data Visualization

ANDREAS BUJA<sup>1</sup> DIANNE COOK<sup>2</sup>,  
DANIEL ASIMOV<sup>3</sup>, CATHERINE HURLEY<sup>4</sup>

March 31, 2004

Projections are a common tool for dimension reduction and visualization of high-dimensional data. The generic example is a projection from  $\mathbb{R}^p$  down to  $\mathbb{R}^2$  which can be used to generate a 2-D view of  $p$ -D data. Going beyond static projections, tools have been developed over the last two decades for interactive viewing of high-dimensional data with *dynamic*, that is, moving or animated projections. These can be metaphorically described as moving film cameras in high-dimensional Euclidean spaces that return movies consisting of dimensionally reduced views. Such tools are variously known as “grand tours”, “guided tours”, and “manual tours”. A special case are “rotations” in 3-D spaces.

The purpose of this article is to describe dynamic projections in mathematical terms, and to show how their differential geometry links up with very concrete issues of animated views of data. Differential geometry applies because dynamic projections are just curves of projections.

We start by discussing graphical rendering methods for projections of data, including cases in which the projection dimension is larger than one, two or three. We continue by examining the kinematics of dynamic projections in terms of invariant Riemannian metrics on Stiefel and Grassmann manifolds. We introduce the notions of “pure plane motion” and “within-plane spin” which are essential for linking the differential geometry of these manifolds to their concrete meaning in terms of animated views of high-dimensional data. The first mathematical main result is a structure theorem that describes all rotation-invariant Riemannian metrics on Stiefel manifolds as Pythagorean mixtures of pure plane motion and within-plane spin. The second main result describes all geodesics with regard to all rotation invariant Riemannian metrics on Stiefel manifolds.

---

<sup>1</sup>Statistics Department, The Wharton School, University of Pennsylvania, 471 Huntsman Hall, Philadelphia, PA 19104-6302; <http://www-stat.wharton.upenn.edu/~buja/>

<sup>2</sup>Dept of Statistics, Iowa State University, Ames, IA 50011; [dicook@iastate.edu](mailto:dicook@iastate.edu), <http://www.public.iastate.edu/~dicook/>

<sup>3</sup>Mathematics Department, University of California, Berkeley, CA 94720; [asimovd@aol.com](mailto:asimovd@aol.com).

<sup>4</sup>Mathematics Department, National University of Ireland, Maynooth Co. Kildare, Ireland; [churley@maths.may.ie](mailto:churley@maths.may.ie), <http://www.maths.may.ie/staff/churley/churley.html>

# 1 Introduction

## 1.1 Projections from $\mathbb{R}^p$ to $\mathbb{R}^d$

This article attempts to lay the mathematical foundations of techniques for the visual exploration of spaces of more than three dimensions with dynamic (animated, moving) projections. The dimension onto which the data are projected is permitted to be not only one or two, but any dimension up to the full dimension of the data. The concepts we introduce are very different from anything taught in conventional 3-D computer graphics, hence the groundwork to be laid is extensive. We start by discussing the problem of graphical rendering of projections (Section 3) and explaining how projections onto arbitrary dimensions can be meaningful and useful: as usual a 2-dimensional projection may be rendered by a plain scatterplot, but a projection onto 5 dimensions may be rendered by a parallel coordinate plot or a scatterplot matrix with five axes; in the extreme, the plot may render a fully  $p$ -dimensional rotation of all of data space as in ExploreN (Carr, Wegman, and Luo 1996) and CrystalVision (Wegman 2003). In Section 3 we give an incomplete list of rendering methods in relation to the projection dimension.

For the interpretation of a view, rendering the projected data alone is not sufficient — it is equally necessary to convey information about the position of the projection in data space. A simple example for scatterplots of 2-D projections of 3-D data is the projected tripod representing the three coordinate axes. Generalizations of tripods to other graphical rendering methods are described in Section 4.

## 1.2 Frames versus planes: the role of orientation in the projection

From graphical rendering considerations we find that the notion of projection is ambiguous. Is the result of a projection an oriented image, or are all orientations of a projection image equivalent? This question leads to the distinction between projection onto a frame — which imposes a particular orientation — and projection onto a plane — which abstracts from particular orientations. The distinction is important in practice: For some types of renderings, orientation is essential (parallel coordinate plots), while for others it is arguably irrelevant (scatterplots). These issues and some of their implications for interpolation of projections are discussed in Sections 5 and 6.

The distinction between projection onto frames and projection onto planes becomes even more important when we consider the kinematics of dynamic projections. We will find that frame motion can be decomposed into pure plane motion — which rotates the plane straight away from itself — and within-plane spin or “whip” spin for short — which rotates the plane within itself. We analyze the relation between these two types of motion in Section 7. In the case of renderings for which orientation is irrelevant, whip spin is generally undesirable, and optimal plane motion should avoid whip spin. For full-dimensional data rotations, however, whip spin is the substance.

The ideas of pure plane motion and whip spin help us analyze the information content in motion. It turns out that motion displays additional data projections, al-

though not as effectively as location. The exact meaning of this statement is explained in Section 8.

### 1.3 Mathematical results for Stiefel manifolds

In the most mathematical part of this work, Sections 9 and 10, we examine the notions of distance and speed of frames and planes. Speed is related to distance because it is just distance traveled per time unit. A priori it is not clear how to measure distance of frames and planes, although proposals can be made up easily. We systematize the question by considering all possible Riemannian metrics on projection frames, introducing some intuitive invariance requirements, and showing that essentially only a one-parameter family of distance measures qualifies. In concrete terms, the result says that there exists essentially only one way to measure speed of pure plane motion and only one way to measure speed of whip spin, but one can choose how to combine the two speed components into an overall speed measure of frames. The resulting ambiguity has consequences: Depending on the choice of speed measure, one obtains different optimal (straightest, geodesic) paths connecting two frames. “Optimal” means roughly that at fixed speed the shortest time is taken. Fortunately, the ambiguity does not exist for paths of frames that consist of pure plane motion or of pure whip spin, which, it turns out, are the two most important cases for our construction.

In the appendix (Section 12) we develop the differential geometry that underlies the theorems used in Sections 7 and 9.

The notions of speed and hence distance are very relevant in actual software implementations of dynamic projections. A viewer should be allowed to interactively choose a speed, and thereafter be exposed to constant speed of motion. As we have seen there is a certain arbitrariness in how speed is measured, but we give arguments in favor of two particular choices: weighting pure plane motion and whip spin equally, or weighting the former twice as much as the latter (Section 9.4).

We also wish to point out that geodesic motion is of practical relevance: it underlies the implementations of dynamic projections in two software systems, GGobi and XGobi.

### 1.4 References to software, algorithms and applications

This paper is about mathematical aspects of dynamic projections. In a companion paper (Buja et al. 2004) we describe algorithmic aspects for implementing dynamic projections. Related as well as very different, namely fractal-based algorithms are discussed by Wegman and Solka (2002).

Free software that implements dynamic projections can be obtained as follows:

- **GGobi** by Swayne, Temple-Lang, Cook, and Buja for Linux and MS Windows<sup>TM</sup>:

<http://www.ggobi.org/>

See Swayne, Buja, and Temple-Lang (2003).

- **XGobi** by Swayne, Cook and Buja (1998) for Unix<sup>®</sup> and Linux operating systems:

<http://www.research.att.com/areas/stat/xgobi/>

See Swayne, Cook and Buja (1998). A version that runs under MS Windows<sup>™</sup> using a commercial X<sup>™</sup> emulator has been kindly provided by Brian Ripley:

<http://www.stats.ox.ac.uk/pub/SWin/>

- **CrystalVision** for MS Windows<sup>™</sup> by Wegman, Luo and Fu

<ftp://www.galaxy.gmu.edu/pub/software/>

See Wegman (2003).

- **ExploreN** for SGI Unix<sup>®</sup> by Luo, Wegman, Carr and Shen:

<ftp://www.galaxy.gmu.edu/pub/software/>

See Carr, Wegman, and Luo (1996).

- **Lisp-Stat** by Tierney contains a grand tour implementation:

<http://lib.stat.cmu.edu/xlispstat/>

See Tierney (1990, chapter 10).

Examples of applications and data analysis can be found in the following articles: Buja, Cook and Swayne (1996); Cook, Buja, Cabrera and Hurley (1995); Furnas and Buja (1994); Hurley and Buja (1990), Wegman (1991)); Wegman and Carr (1993); Wegman and Luo (1996); Wegman and Shen (1993); Symanzik, Wegman, Braverman, and Luo (2002); Wegman, Poston, and Solka (1998); Wegman (2003)

It should be kept in mind, though, that the printed paper has never been a satisfactory medium for conveying intuitions about motion graphics. Nothing replaces live or taped demonstrations or best of all: hands-on experience.

## 2 Preliminaries

**Conventions:** We use the term “projection plane” or simply “plane” in a very general sense, meaning a subspace of arbitrary dimension  $d$  in  $p$ -dimensional data space. We consider projection dimensions  $d$  anywhere between 1 and the dimension  $p$  of data space. In XGobi and GGobi,  $d = 2$  and  $d = 1$  are the typical dimensions, but for Wegman (1991)  $d = p$  is generic. The “data dimension”  $p$  really denotes the number of currently active variables, as opposed to the number of *all* variables. This distinction is necessary in interactive systems such as X/GGobi, ExploreN and CrystalVision, where arbitrary subsets of the variables can be interactively chosen for viewing with dynamic projections.

**Remark on metrics in data space:** We will always assume that data space is equipped with a canonical inner product that determines a length unit across variables and assumes variable directions to be orthogonal. We are not concerned with the problem of choosing an inner product but simply assume that some choice has been made. It should be up to the data visualization system to provide various choices by allowing viewers to choose different scalings of the data. In XGobi and GGobi, for example, viewers can choose the half-range, the standard deviation, or the median absolute deviation (MAD) as the unit length.

**Notation:** A  $d$ -dimensional projection plane is generated by an orthonormal basis which we collect in a  $p \times d$ -matrix  $F = (\mathbf{f}_1, \dots, \mathbf{f}_d)$  called a  **$d$ -frame**. Note that  $F^T F = I_d$  due to orthonormality of the columns. We denote the plane by  $\text{span}(F)$ . Any plane has of course infinitely many frames that span it. The projection of a high-dimensional data vector  $\mathbf{x}_i$  onto the direction  $\mathbf{f}_j$  is given by the scalar product  $\mathbf{f}_j^T \mathbf{x}_i$ . The projection of  $\mathbf{x}_i$  onto the orthonormal frame  $F$  is the  $d$ -dimensional vector

$$\text{View}_i = F^T \mathbf{x}_i .$$

To make projections dynamic, they have to be functions of a time parameter  $t$ :

$$F(t) = (\mathbf{f}_1(t), \dots, \mathbf{f}_d(t)) ,$$

and same for the projected data:

$$\text{View}_i(t) = F(t)^T \mathbf{x}_i$$

### 3 Graphical Rendering of a Projection

A rendering method is a way of creating a graphical scene from a  $d$ -dimensional projection. In this sense, a scatterplot is a rendering of a 2-D projection, but a parallel coordinate plot with  $d$  axes is equally a rendering of a  $d$ -dimensional projection.

Rendering methodology is a wide area, covering much of statistical data visualization; see Wegman and Carr (1993) for an excellent introduction. In what follows we describe a few methods that are suitable for rendering real-time dynamic projections of multivariate data given widely available display technology. Most have seen actual use in dynamic graphics systems. We limit ourselves to inexpensive graphical scenes that consist of points and lines, such as pointclouds, polygons, curves and wireframes. This graphical vocabulary is sufficiently rich to permit renderings of data projections as scatterplots, time series plots, simple geometric surfaces and geometric bodies and graphs (see Littman et al. 1992 for visualizing high-dimensional graph layouts).

We are not concerned with the typical rendering questions that are at the heart of most of 3-D computer graphics, such as lighting models and hidden line removal, much of which is dependent on the presence of a single back-dimension. See again Wegman and Carr (1993) for such issues. For some interesting higher-dimensional

rendering attempts, see Hanson and Heng (1991) who propose a 4-D shading method, and Young and Rheingans (1991) who experiment with high-dimensional depth-cuing.

One type of complex rendering, however, should be mentioned: Scott (1992, 1995) reminds us that density displays in 2 and 3 dimensions are valuable for large data sets that cause extensive overplotting in conventional 2-D scatterplots and obscuring in 3-D point scatters. Similarly, Carr (Carr et al. 1986, Carr 1991) has argued in favor of binned displays for large data samples. In a different context, Miller and Wegman (1991) explore density enhancements for parallel coordinate plots. Some of this is now in ExploreN with hardware support based on alpha-blending techniques (Carr et al. 1996).

We discuss rendering methods for projected multivariate data in the order of the projection dimension.

### 3.1 Rendering of 1-dimensional projections

The obvious method of displaying a 1-D projection as points along a line is not useful due to the problem of overstrike. One way out of this problem is to make meaningful use of the second screen dimension. Below we describe several options, all of them illustrated in Figure 1. Other possibilities, not discussed here, are aggregating displays such as histograms and boxplots.

- **Jitter plots:** A simple method to alleviate the overstrike problem is to spread out the points in the second screen dimension, for example with random “jitters”, by plotting the 1-D projection against random numbers:

$$Hor_i = S_H \cdot Random_i + C_H, \quad Vert_i = S_V \cdot \mathbf{f}_1^T \mathbf{x}_i + C_V,$$

where  $S_H, S_V, C_H, C_V$  are horizontal and vertical scale and centering constants, respectively. Perceptionally superior to simple random jitters is a sophisticated method by Tukey and Tukey (1990) called **textured dot plots**. They create jitters that are slightly more regular so as to avoid the random clumps that come with the use of simple random numbers. Textured dot plots are computationally more expensive than random number jitters because the jitters are computed based on the distribution of the 1-D projection. For dynamic projections this implies continuous updating as the projection moves in time.

- **Density plots:** Another use of the second screen dimension is by plotting an estimated density function  $\hat{p}$  against the projection:

$$Hor_i = S_H \cdot \mathbf{f}_1^T \mathbf{x}_i + C_H, \quad Vert_i = S_V \cdot \hat{p}(\mathbf{f}_1^T \mathbf{x}_i) + C_V.$$

An early use of dynamic density plots appeared in the PRIM-H system (Donoho, Huber, Ramos and Thoma 1982). Tierney (1990) demonstrated a histogram tour in his Lisp-Stat system. In Hurley and Buja (1990) we used average shifted histograms for computationally fast density estimation (Scott 1985), which was

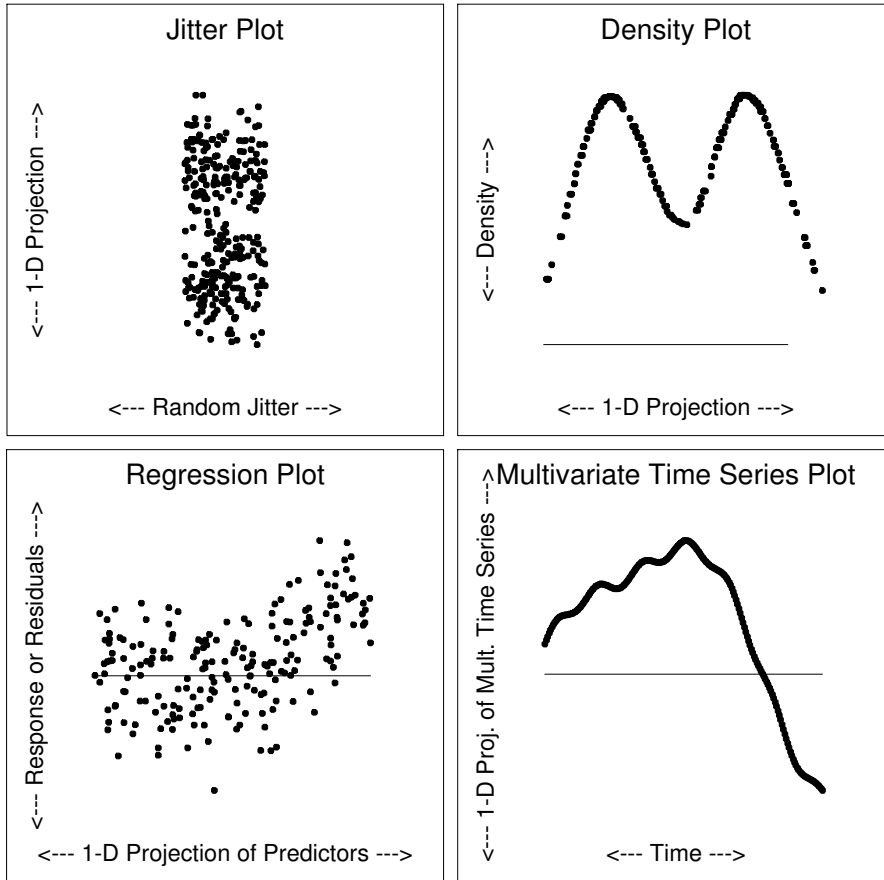


Figure 1: *Examples of 1-D rendering methods: jitter plots, density plots, regression plots, and multivariate time series plots.*

replicated in X/GGobi. Both for speed and for the ability to point at individual cases in the data, we only plotted the density values at the projected data points. Even if overstrike occurs, the presence of high density areas in the projection is conveyed graphically.

- **Regression plots:** A useful application of dynamic 1-D projections is to predictor variables in regression problems. We plot a fixed variable  $y_i$ , such as a response or (partial) residual, against the dynamic 1-dimensional projection of some of the predictors  $\mathbf{x}_i$ :

$$Hor_i = S_H \cdot \mathbf{f}_1^T \mathbf{x}_i + C_H, \quad Vert_i = S_V \cdot y_i + C_V.$$

A thorough search over  $\mathbf{f}_1 = \mathbf{f}_1(t)$  may lead to the discovery of nonlinearities, heterogeneous variance, or extreme response values (Cook and Weisberg 1994).

- **Time series plots:** We have also found it useful to plot the dynamic projection  $\mathbf{f}_1^T \mathbf{x}_i$  vertically and use an auxiliary fixed variable  $x_i$  as the horizontal axis:

$$Hor_i = S_H \cdot x_i + C_H , \quad Vert_i = S_V \cdot \mathbf{f}_1^T \mathbf{x}_i + C_V .$$

We have applied this technique to multivariate time series data  $\mathbf{x}_i$ , in which case it is sensible to use time as the horizontal auxiliary variable  $x_i$ . (Time refers here to observed time in the data, not the time parameter  $t$  of the dynamic projection vector  $\mathbf{f}_1(t)$ .)

The last two rendering techniques can also be interpreted as particular 2-D projections of the data, but the dynamic part is only a 1-D projection. Both techniques were tried out in the DataViewer system (Buja et al. 1988) as special cases of a broader class of methods called “correlation tours.” These are now available in XGobi and GGobi.

An application of 1-D tours to multi-spectral images is described in Wegman, Solka and Poston (1996): They use the linear combination produced by a restricted 1-D tour for dynamic grey scale coding of 6-dimensional images.

### 3.2 Rendering of 2-dimensional projections

The usual way of mapping a projection 2-plane to a scene is by assigning the vectors of the 2-frame  $F = (\mathbf{f}_1, \mathbf{f}_2)$  to the horizontal and vertical screen dimensions, respectively:

$$Hor_i = S_H \cdot \mathbf{f}_1^T \mathbf{x}_i + C_H , \quad Vert_i = S_V \cdot \mathbf{f}_2^T \mathbf{x}_i + C_V ,$$

which results in the usual Cartesian scatterplot, possibly enhanced with lines for interpretability. It may be sensible to choose  $S_H$  and  $S_V$  such that one length unit in the projection plane translates to the same physical dimensions horizontally and vertically on the computer screen. This amounts to  $S_V = -S_H$  if the physical dimensions of the pixels are identical horizontally and vertically, and if the hardware uses the conventional axis orientation (left to right and top to bottom).

### 3.3 Rendering of 3-dimensional projections

For a 3-dimensional projection with corresponding 3-frame  $(\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3)$ , three common viewing methods are: Stereo views, depth cues, and 3-D rotations. The most powerful 3-D views take advantage of all three methods simultaneously. In the following we assign the frame vectors to horizontal, vertical, and depth directions, respectively.

- **Stereo views**, in the simplest case, can be obtained by generating two sets of screen coordinates that differ slightly in their 2-dimensional views, one view per eye:

$$Hor_i = S_H \cdot (\cos \eta \cdot \mathbf{f}_1 + \sin \eta \cdot \mathbf{f}_3)^T \mathbf{x}_i + C_H , \quad Vert_i = S_V \cdot \mathbf{f}_2^T \mathbf{x}_i + C_V ,$$



where the exposure angle  $\eta$  may be  $-1^\circ$  for the left eye and  $+1^\circ$  for the right eye, if we interpret the  $\mathbf{f}_3$ -direction as pointing away from the viewer (see Wegman and Carr (1993), Section 6, for a detailed discussion). Separate exposure for each eye can be achieved by several techniques: 1) special purpose screens that generate two overlaid images in different polarizations, combined with suitable glasses for the viewer to achieve separate exposure of the eyes to the two images; 2) special purpose mirrors that match two distinct screen windows with separate projections to the left and the right eye; 3) alternating exposure of the two eyes to separate images at rapid speed; 4) general purpose color screens on which two overlaid images are generated in, say, blue and red colors, combined with red-blue glasses to achieve separate eye exposure. Among these possibilities, the last is low-tech and does not require extra hardware besides cheap red-blue glasses. It is limiting, however, in that it confines the implementor to one specific use of color. The above simple formulae are sufficient for creating the stereo effect but if desired one could follow standard computer graphics techniques and implement sophisticated 3-dimensional perspective viewing with finite eye position.

- **Depth cues** or, strictly speaking, monocular depth cues (in contrast to binocular depth cues provided by stereo views): These are created by forcing certain viewing parameters of the display objects to be monotone functions of the back projection

$$Depth_i = S_D \cdot \mathbf{f}_3^T \mathbf{x}_i + C_D .$$

Examples of such viewing parameters are:

- Brightness: Show far objects dimmer than near objects,
- Blue-tint or fog: Show far objects as seen through the atmosphere over a distance,
- Size: Draw far objects smaller.

See Figure 2 for examples of brightness and size cues.

- **3-D rotations** are generated as dynamic images, in the simplest case of a rotation around the vertical axis, by

$$Hor_i = S_H \cdot (\cos t \cdot \mathbf{f}_1 + \sin t \cdot \mathbf{f}_3)^T \mathbf{x}_i + C_H , \quad Vert_i = S_V \cdot \mathbf{f}_2^T \mathbf{x}_i + C_V ,$$

where  $t$  is a time parameter.

It might appear that 3-D rotations are not suitable for rendering dynamic 3-D projections because such rotations are dynamic themselves. However, an early proposal by Tukey and Tukey (1981, p. 272-3) suggests just this: In essence, they would run a grand tour with  $d = 3$  at slow speed and spin the 3-D projection space within itself at high speed, a process they call **wabing**. They derive this term from the first stanza in Lewis Carroll's "Jabberwocky" (from "Through the Looking Glass"):

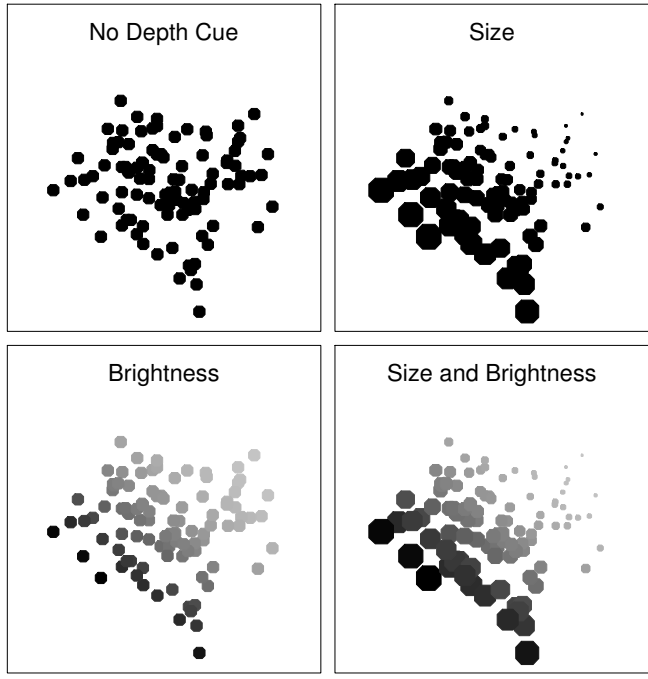


Figure 2: *Examples of depth cues for 3-D rendering methods: size and brightness.*

'Twas brillig, and the slithy toves  
 Did gyre and gimble in the wabe;  
 All mimsy were the borogoves,  
 And the mome raths outgrabe.

Borrowing further terms from “Jabberwocky”, they call the fast 3-D rotation **gyring** and the slow tour motion **gimbling**. This would allow one to take advantage of 3-D perception without stereo technology and yet move through higher-dimensional data space at the same time. The drawback of wabing is that the speed of the tour motion has to be relatively slow, but unlike some of the depth cue techniques, there is no interference with the use of color.

An implementation reminiscent of wabing is available in Tierney’s (1990) Lisp-Stat system: His 2-D tour carries out a full-dimensional tour but shows only two dimensions while in progress. When the tour is stopped, a third dimension is accessible for interactive 3-D rotation in the familiar manner.

### 3.4 Rendering of projections onto dimensions greater than 3

We consider projection onto dimensions greater than 3. In the extreme, the projection dimension  $d$  can equal the dimension  $p$  of data space, in which case the dynamic projection amounts to a full display of all variables in a dynamically rotating coordinate system. Some examples of  $d$ -D rendering methods follow and these are illustrated

in Figure 3:

- **Scatterplot matrices** (called “generalized draughtman’s views” by Tukey and Tukey (1981, p. 206f); see also Chambers et al. (1983)): One arranges the scatterplots of all variable pairs in a matrix arrangement. If used as a rendering method of a  $d$ -dimensional projection, the  $(j, k)$ ’th scatterplot of the  $d \times d$  matrix arises from the projections onto the frame vectors  $\mathbf{f}_j$  and  $\mathbf{f}_k$ . This method is used in the ExploreN system (Carr et al. 1996) to render a full-dimensional grand tour with  $d = p$ .
- **Parallel coordinate plots** (Inselberg 1985, Wegman 1991, Wegman and Luo 1996): One uses parallel lines to represent the  $d$  coordinate axes; one then represents a case by a polyline that connects the coordinate values on the axes with straight line segments. This method depends on an ordering of the coordinates, and it generally works best for small numbers of cases. In order to render a projection, the frame vector  $\mathbf{f}_j$  yields the  $j$ ’th axis of the display. Wegman (1991) and Wegman and Luo (1996) illustrate the use of parallel coordinates for a full-dimensional tour with  $d = p$ . It is also implemented in the ExploreN system (Carr et al. 1996).
- **Andrews curves** (Andrews 1972): The description of this technique is often buried under arcane details involving trigonometric functions. The essential idea is this: Each case is represented by a curve that is obtained as a trace of smoothly varying 1-D projections: Case  $\mathbf{x}_i$  is represented by the curve  $s \rightarrow \mathbf{f}(s)^T \mathbf{x}_i$ , where  $\mathbf{f}(s)$  is a 1-parameter family of projection vectors. Effectively, Andrews curves display 1-D tours as plots of functions of the time parameter, as opposed to the conventional 1-D tour which makes use of the time parameter as physical time. If used as a rendering method of dynamic projections, Andrews curves will be dynamically computed from  $View_i(t) = F(t)^T \mathbf{x}_i$  rather than  $\mathbf{x}_i$ . As of this writing we do not know of a tour implementation that makes use of Andrews curves for rendering dynamic projections.

Other rendering methods for projections onto  $d$  dimensions exist (generally of the glyph type: stars, castles, ...) and could be used for tour rendering as well.

## 4 Knowing Where We’re Looking in High-Dimensional Space: Rendering Frames

A dizzy feeling besets many first-time viewers of high-dimensional data projections, and they rightfully ask “How do I know what I’m looking at?” Crucial for the successful use of data projections is the availability of tools for interpreting views. In geometric terms, the task supported by such tools is that of locating the position of the projection frame in  $p$ -space. In numeric terms, the task really boils down to interpreting the numbers contained in the  $p \times d$  frame  $F$ . In principle, one could simply

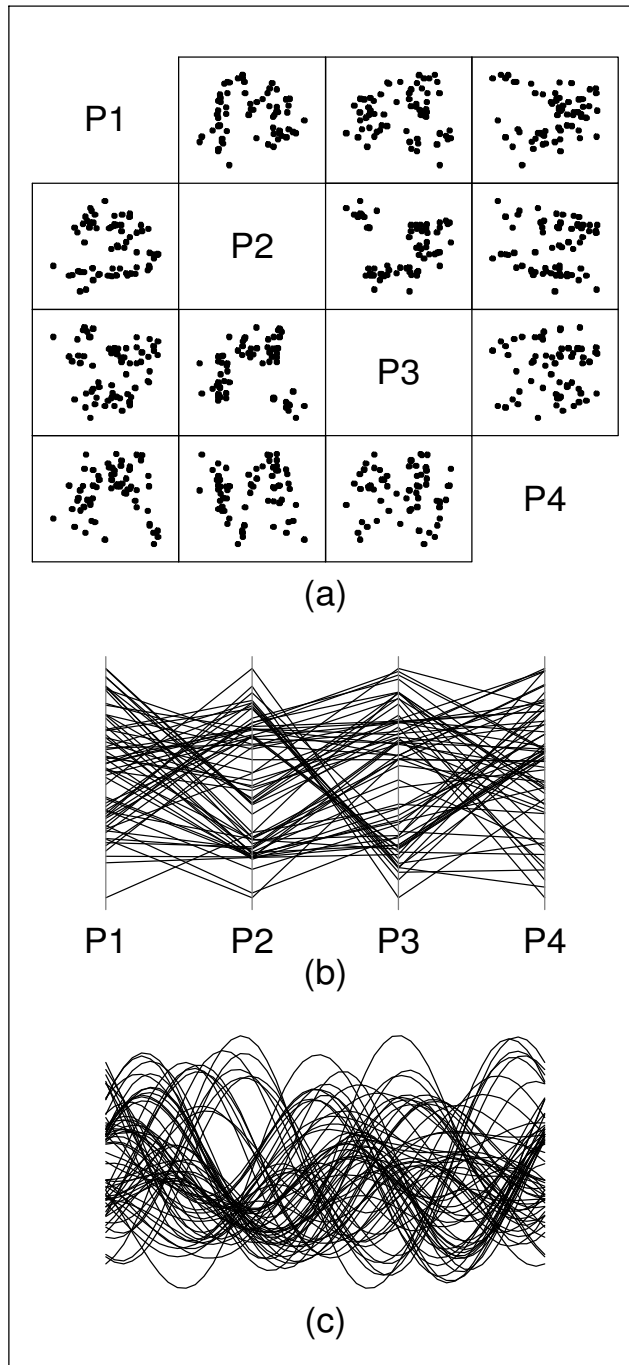


Figure 3: *Examples of  $d$ -D rendering methods for  $d = 4$ : (a) Scatterplot matrix, (b) parallel coordinates, (c) Andrews curves. The data are the first four principal components extracted from some speech data representing 11 vowels, each vowel replicated 6 times ( $N = 66$ ). Thus, 11 groups of size 6 are known to exist, which may or may not show up as clusters in the rendered projections.*

print these numbers as a matrix, and this may be useful at times: The values show how much individual variables “load” on the 1-D projections given by the columns  $\mathbf{f}_j$  of  $F$ . There exist, however, better and more visual ways of conveying this information.

We take a general view and ask how arbitrary rendering methods can be supplemented with tools for visually rendering  $F$ . In particular, the principles should apply to all methods discussed in the previous section: 1-D renderings with jitter plots, density plots, regression plots, multivariate time series plots; 2-D renderings with scatterplots; 3-D renderings with stereo views, depth cues, 3-D rotations;  $p$ -D renderings with parallel coordinate plots, scatterplot matrices, Andrews curve plots.

Two rendering principles that often lead to reasonable solutions are the following:

1. Render the  $d$  column vectors of  $F$  individually with graphical renditions of the  $p$  numbers in each column. That is, make  $d$  times use of something that works for rendering a 1-D projection vector.
2. Interpret the variable unit vectors in  $p$ -space as regular data, project them with  $F$  like the real data, and render the result also like the real data. That is, render the rows of  $F$  as if they were projected data.

An example of rendering a 1-D projection frame is shown in the lower part of Figure 4: The  $p$  numbers in the 1-frame are plotted below the data area and coded as values on a common scale on vertically shifted axes in order to avoid overplotting. One easily reads off how much and with which sign each variable contributes to the projection. — This method can be easily generalized according to the first principle: In parallel coordinate plots and scatterplot matrices, for example, one could attach a suitably miniaturized version of these 1-D renditions to each of the  $d$  projection dimensions, sideways or below the horizontal or vertical axes.

An example of the application of the second principle is shown in Figure 5: The traditional tripod of 3-D computer graphics is just an enhanced rendition of the projection of the three variable unit vectors. Similarly, the generalized tripod called “ $p$ -pod” shown when touring in X/GGobi is an enhanced rendition of the  $p$  variable unit vectors in  $p$ -space. In X/GGobi, the  $p$ -pod is also rendered by a second method in order to avoid overplotting and to provide interactive manipulation of the variables, see the right of the same figure. — The  $p$ -pod method works also for 3-D projections rendered with 3-D stereo views and 3-D rotations: The pod will look like a star with  $p$  unequal rays in 3-space.

The second principle can be applied to parallel coordinate plots and Andrews curves: Treat the variable unit vectors as if they were data, render them with polylines or curves, and place, mark and label them visually so they are recognized as guide posts rather than data. For an example in terms of Andrews curves, see Figure 6.

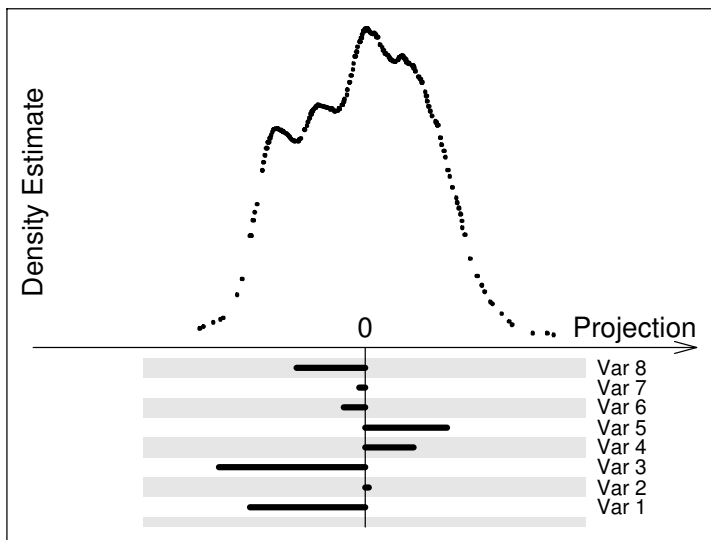


Figure 4: *Rendering of the current frame position for 1-D projections: The example shows  $p = 8$ . The data are rendered with a density plot, while the position of the projection direction is rendered with horizontal bars below the plotting area.*

## 5 When do we Project on a Frame, When on a Plane?

### 5.1 Orientation: Why it matters

Section 3 makes it clear that the notion of projection is quite ambiguous: For  $d = 2$  one might assume that the essential substrate of the projection is the plane  $\text{span}(F)$ , but for  $d = p$  the projection subspace is trivial, namely all of data space, and the essential substrate of the projection is not the space but its orientation as given by the full  $p \times p$  frame  $F$ .

The questions then are: When does orientation matter, and when are all orientations of a plane equivalent for visualization purposes? Does it matter whether an axis is used left-to-right or right-to-left, whether a scatterplot is shown 45 or 90 degrees rotated, which among three variables is coded as depth cue in a 3-D scatterplot, whether the axes in a parallel coordinate plot are permuted or replaced by linear combinations of the original variables? Generally, when are a frame  $F$  and a rotated version  $FV$  thereof equivalent? ( $V$  is  $d \times d$  and  $V^T V = I_d$ .)

Some of these questions concern visual perception and should be answered by empirical research. In the absence of such research, we are left with a priori reasonings that hopefully will not be too far off. The role of orientation generally depends on

- the rendering method and
- the type of scenes we expect to see.

In the next section we give our best guesses for the rendering methods described in the previous section for dimensions  $d = 1, 2, 3$ , and  $d > 3$ .

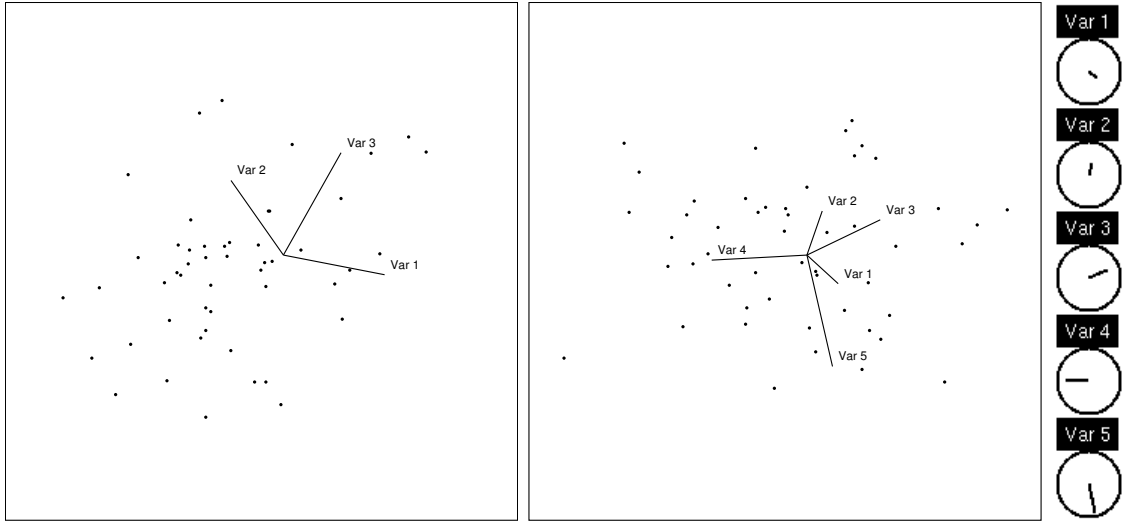


Figure 5: *Rendering of the current frame position for scatterplots of 2-D projections: Left: A conventional tripod representing the projections of the unit vectors of the three variables in 3-D data space.*

*Right: A generalization of the tripod to a so-called “p-pod” is shown in the plotting area. In this example  $p = 5$ . On the right is the X/GGobi-style rendition of the p-pod: The rays of the p-pod are disentangled and placed in individual variable circles to avoid overplotting and allow manipulation of variables with mouse clicks. The rays again represent the projections of the variable unit vectors onto the current projection plane. Equivalently, the  $j$ 'th ray represents the  $j$ 'th row vector of the 2-frame  $F$ .*

Our interest in the question of orientation stems from the fact that it has fundamental implications for tour interpolations:

- If projections are rendered such that orientation is irrelevant, we need to interpolate planes  $\text{span}(F)$ . It is then possible to establish precise optimality of certain interpolating paths of planes.
- If projections are rendered such that orientation matters, we need to interpolate frames  $F$  rather than planes. Curiously, this case is theoretically less tractable, with one exception: In the extreme, when the plane  $\text{span}(F)$  is at rest and the dynamic frame  $F$  is essentially a dynamic basis of  $\text{span}(F)$ , optimal interpolations do exist. This includes the case of full-dimensional tours  $p = d$ .
- For general dynamic frame motion we will see that even the criteria for optimality are not unique. The reason is that no unique method for measuring speed of frame motion exists (Section 9).

## 5.2 Orientation: When it matters

In what follows, it should be kept in mind that every rendering method depends on roles assigned to the vectors of a basis frame  $F = (\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_d)$ . In a 3-D scatterplot,

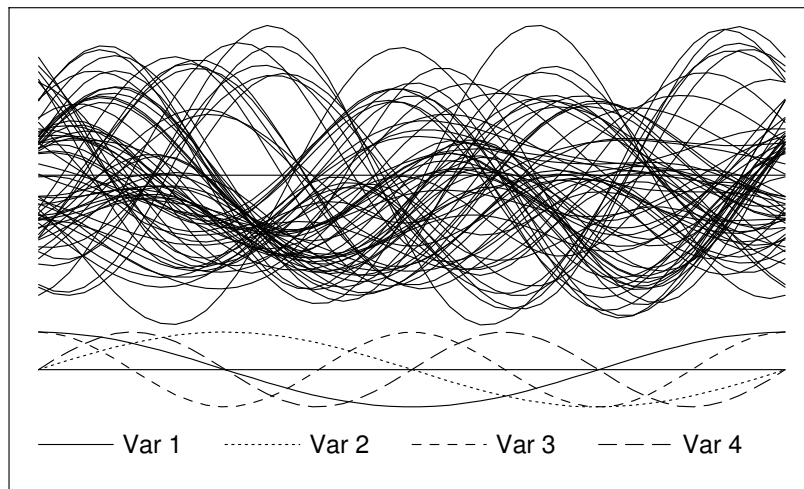


Figure 6: *Rendering of the current frame position for Andrews curves: The upper curves represent the data, the lower and smaller curves represent the variable unit vectors. They express how much each variable contributes at each location of the horizontal axis.*

for example,  $\mathbf{f}_1$  may be used for the horizontal screen direction,  $\mathbf{f}_2$  for the vertical direction, and  $\mathbf{f}_3$  for the depth cues. Here is a discussion of the role of orientation for each of the rendering methods of Section 3:

**$d = 1$ :** In all applicable rendering methods, the single projection dimension is used as a horizontal or vertical screen axis. Irrelevance of orientation means that a projection can be equally displayed left-to-right and right-to-left when used horizontally, or bottom-up and top-down when used vertically. When the projection is fixed and has a specific meaning, such as time, orientation may matter. This can be due to the nature of the visual scene: Text, for example, is hard to read when mirror-imaged. In the kinds of scenes we expect from dynamic 1-D projections of multivariate data, however, no specific interpretation of the projection exists other than being a linear combination of variables, and the scenes are not structured in a way that makes a distinction between left-to-right and right-to-left essential. Therefore, orientation seems irrelevant for dynamic 1-D projections.

**$d = 2$ :** The only case we consider is essentially a scatterplot of the two 1-D projections. Irrelevance of orientation means that rotated and axially reflected scatterplots are visually equivalent. As for  $d = 1$ , there do exist scatterplots where orientation matters because of the nature of the scene and the specific interpretations of the axes. An example are again time series, but so are all plots whose main feature of interest is slope. The kinds of plots we expect from dynamic 2-D projections of multivari-



ate pointclouds, however, would rarely have features whose perception depends on a specific orientation on the screen. Typical structure such as clusters, lines, curves, outliers and combinations thereof can be recognized quite easily without turning a scene around. Therefore, orientation seems irrelevant for dynamic 2-D projections.

**$d = 3$ :** In this situation, the rendering methods discussed above show somewhat different behaviors under changes of orientation. Depth cues and stereo views do not provide as much resolution in the back dimension as does location in the two front dimensions. Therefore, orientation matters somewhat. It matters less for 3-D rotations: The depth dimension changes continuously, and what was depth is screen location a moment later after a 90 degree rotation. This is true for Tukey and Tukey’s (1981) wabing proposal for rendering a slowly moving  $p$ -to-3-D projection with fast moving 3-D rotations. This rotation amounts to a purposeful use of within-plane spin for rendering. It is therefore desirable that this 3-D rotation be unaffected by the motion of the 3-frame, which just means that the moving 3-frame be free of within-plane spin.

**$d > 3$ :** In more than 3 dimensions, the three rendering methods discussed above — scatter plot matrices, parallel coordinate displays and Andrews curves — all depend on orientation through their particular use of the projection directions  $\mathbf{f}_1, \dots, \mathbf{f}_d$ . We do not know of any methods for orientation-independent rendering in this general case. For a full-dimensional tour  $d = p$ , the point of a tour is to expose the viewer to different orientations of full-dimensional data space, as opposed to different low-dimensional projections. Therefore, underlying a full-dimensional tour is the assumption that data space is rendered with essential use of orientation.

The answer to the question of whether to project onto a plane or a frame is primarily a function of the rendering method. The choice of rendering method, however, is largely a function of the projection dimension, hence the question of planes versus frames ends up being a function of the projection dimension  $d$  by proxy. In summary:

- When rendering with scatterplot methods onto dimensions 1, 2 and 3, we project onto a *plane*;
- when rendering with scatterplot matrices, parallel coordinate plots or Andrews curves onto dimensions  $d > 3$ , we project onto a *frame*.

## 6 Interpolating Frames and Planes

For visualization of data projections it is always necessary to refer to a frame

$$F = (\mathbf{f}_1, \dots, \mathbf{f}_d)$$

because some orientation of the projection is needed for rendering. This holds even when orientation is irrelevant: Irrelevance only means that renderings that differ in orientation are perceptually equivalent, not identical.

When orientation matters for a rendering method, a frame is the essential substrate: A frame  $F$  and a rotated version  $FV$  of the same frame do *not* produce equivalent visual scenes, although the planes are the same:  $\text{span}(F) = \text{span}(FV)$ . [ $V$  is  $d \times d$  and orthogonal  $V^T V = I_d$ .]

When orientation does not matter, rendering is still based on a frame  $F$ , but rotated versions  $FV$  of the frame yield equivalent visual scenes. We therefore need a mathematical substrate that encodes this equivalence. The most plausible substrate of a plane,  $\text{span}(F)$ , is mathematically inconvenient. More useful is the orthogonal projection matrix  $P$  generated by  $F$ :

$$P = FF^T .$$

The planes of dimension  $d$  are in a 1-1 correspondence with the matrices  $P$  characterized by idempotence ( $PP = P$ ), symmetry ( $P^T = P$ ) and rank  $d$ . Obviously,  $F$  and  $FV$  generate the same  $P$ .

**Convention:** *Henceforth the symbols  $F$  and  $P$  will always represent a frame and its associated orthogonal projection matrix.*

We turn to the problem of interpolating frames and planes. Let

$$F_a = (\mathbf{f}_{a,1}, \dots, \mathbf{f}_{a,d}) \quad \text{and} \quad F_z = (\mathbf{f}_{z,1}, \dots, \mathbf{f}_{z,d})$$

be a starting frame and a target frame, respectively. Similarly let

$$P_a = F_a F_a^T \quad \text{and} \quad P_z = F_z F_z^T$$

be the projection matrices that encode the corresponding starting plane and target plane, respectively.

Interpolation of the frames  $F_a$  and  $F_z$  is done with a path of frames  $F(t)$  ( $t_a \leq t \leq t_z$ ) that satisfies

$$F(t_a) = F_a \quad \text{and} \quad F(t_z) = F_z .$$

Interpolation of planes in the abstract means that a path of frames generates a path of projection matrices  $P(t) = F(t)F(t)^T$  that runs from  $P_a$  to  $P_z$ . This definition is not useful in dynamic graphics because the path  $F(t)$  needs to continue from where the previous path left off in order to ensure continuity of motion. Thus, the path needs to satisfy

$$F(t_a) = F_a \quad \text{and} \quad P(t_z) = P_z .$$

The difference between frame and plane interpolation is that in plane interpolation *any frame in the target plane can serve as a target frame*. This confers a certain measure of freedom to plane interpolations. The question is how to take advantage of this freedom. This is one of the topics of the next section: Through proper choice of the target frame, one can avoid undesirable spin within the projection plane.

## 7 Kinematics I: Within-Plane Spin (“Whip Spin”)

### 7.1 Motivation

Consider, for illustration purposes, the canonical (1,2)- and (3,4)-frame in 4-space, and the simplest interpolating path  $F(t)$ :

$$F_a = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad F_z = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad F(t) = \begin{pmatrix} c_t & 0 \\ 0 & c_t \\ s_t & 0 \\ 0 & s_t \end{pmatrix},$$

where we abbreviated  $c_t = \cos(t)$  and  $s_t = \sin(t)$ . The path of 2-frames  $F(t)$  interpolates between  $F_a$  and  $F_z$  when  $t$  runs from 0 to  $\pi/2$ . Now consider an alternative path:

$$\tilde{F}(t) = F(t) \begin{pmatrix} c_{vt} & -s_{vt} \\ s_{vt} & c_{vt} \end{pmatrix}$$

For any value of  $v$ , the paths  $F(t)$  and  $\tilde{F}(t)$  generate the same path of planes, interpolating the (1,2)-plane and the (3,4)-plane. The path  $\tilde{F}(t)$ , however, interpolates the (1,2)-frame and the (3,4)-frame only when  $v$  is an integer multiple of 4:  $v = 4n$ . In an intuitive sense, the most economical interpolation is for  $v = 0$ . If  $v$  is far from zero, the path of frames  $\tilde{F}(t)$  rotates within the plane it spans at great speed, similar to propeller blades of an airplane. In data visualization, where the frames are used to propel projections of high-dimensional objects, the viewer of such projections will perceive an image that spins within the computer screen at high speed, while the change in the image due to the motion of the projection plane itself is obscured. This type of spin is clearly undesirable when plane interpolation is wanted. Because of the importance of this notion we introduce the term **within-plane spin** or contracted **whip spin** to refer to this component of moving frames.

### 7.2 Mathematical definition of whip spin and plane motion

Given a general path of  $d$ -frames

$$F = F(t) = (\mathbf{f}_1(t), \dots, \mathbf{f}_d(t)),$$

denote its time derivative by

$$F' = F'(t) = dF(t)/dt$$

and the time derivative of its columns by

$$\mathbf{f}'_j = \mathbf{f}'_j(t) = d\mathbf{f}_j(t)/dt.$$

[We usually suppress the time parameter  $t$ .]

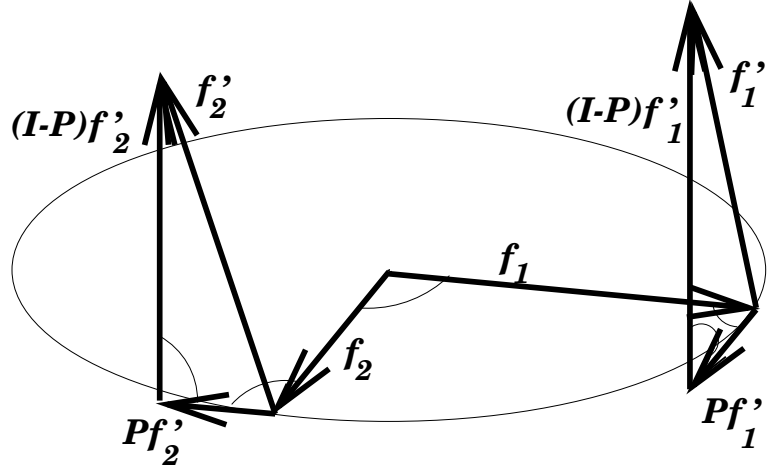


Figure 7: *Decomposition of the derivative  $F'$  of an orthonormal 2-frame  $F = (\mathbf{f}_1, \mathbf{f}_2)$  into whip spin  $PF'$  and plane motion  $(I - P)F'$ . The derivatives  $\mathbf{f}'_1$  and  $\mathbf{f}'_2$  are the tangent vectors of  $\mathbf{f}_1 = \mathbf{f}_1(t)$  and  $\mathbf{f}_2 = \mathbf{f}_2(t)$ , respectively. Note the patterns in the projections:  $P\mathbf{f}'_1 = \alpha \cdot \mathbf{f}_2$  and  $P\mathbf{f}'_2 = -\alpha \cdot \mathbf{f}_1$ , which expresses the geometric meaning of the skew whip spin matrix  $F^T F'$  as an infinitesimal rotation in the plane spanned by  $\mathbf{f}_1$  and  $\mathbf{f}_2$ . (All shown angles are right angles.)*

Whip spin can be quantified by projecting the time derivative of the frame onto the plane spanned by the frame. Plane motion is what's left after removing whip spin. Hence the following terminology:

**Convention:**

- The **whip spin** or **within-plane spin** of  $F$  is  $PF'$ .
- The **plane motion** of  $F$  is  $(I - P)F'$ .

Both are  $p \times d$ -matrices. Whip spin is the component of  $F'$  that takes place within the plane  $\text{span}(F)$ , and plane motion is the component of  $F'$  that shoots vertically out of the projection plane.

Because  $PF' = F(F^T F')$ , all the relevant information about whip spin of  $F$  is contained in the  $d \times d$ -matrix  $F^T F'$ : It describes whip spin with regard to the basis  $F$  of the projection plane. Due to its pervasiveness throughout the rest of the paper we introduce the following:

**Convention:** We call  $F^T F'$  the **whip spin matrix** associated with  $F = F(t)$ .

The  $(j, k)$ -entry of  $F^T F'$  is  $\mathbf{f}_j^T \mathbf{f}'_k$  which represents the projection of  $\mathbf{f}'_k$  onto the direction  $\mathbf{f}_j$ . The following fact is simple but crucial and used everywhere in what follows:

**Fact:** *The whip spin matrix is skew-symmetric for orthonormal paths of frames:*

$$(F^T F')^T = -F^T F' .$$

This is illustrated in Figure 7 for 2-frames in 3-space. The proof is by taking derivatives in the orthonormality condition  $F^T F = I_d$ .

It is easily checked that the path  $F(t)$  in the example of Section 7.1 has zero whip spin:  $F^T F' = 0$ . The path  $\tilde{F}(t)$  has zero whip spin iff  $v = 0$ . It is of interest to generalize this example to arbitrary  $d$ -frames as follows: Let  $F = F(t)$  be a path of  $d$ -frames with zero whip spin; also, let  $V = V(t)$  be a path of  $d \times d$ -rotations. Then the path of frames  $\tilde{F} = FV$  generates the same path of planes as  $F$ , but its whip spin matrix is:

$$\tilde{F}^T \tilde{F}' = V^T V' .$$

That is, the rotations  $V$  are the purveyors of whip spin. [To prove the identity use  $F^T F = I_d$  and  $F^T F' = 0$ .] This example will be used repeatedly in the following sections where we consider the extremes of 0% whip spin and 100% whip spin, respectively.

### 7.3 First extreme: Pure plane motion

Consider the following situation: All the apparent motion is due to motion of the plane. The frame vectors move orthogonally out of the plane. This is characterized by the equivalent conditions

$$F^T F' = 0 , \quad \text{and} \quad P F' = 0 .$$

For a path  $\tilde{F} = FV$  where  $F$  has zero whip spin, the condition is equivalent to  $V^T V' = 0$ . It turns out that any path  $\tilde{F}$  can be represented in this way:

**Theorem 1.** *For every path of frames  $\tilde{F}(t)$ , there exists another path  $F(t)$  that generates the same planes but has zero whip spin. More specifically, there exists a dynamic  $d \times d$ -rotation  $V(t)$  such that*

$$\tilde{F} = FV , \quad F^T F' = 0 , \quad F(0) = \tilde{F}(0) .$$

The proof is given in the Appendix (Section 12.8).

The theorem guarantees that if there exists a path of frames that interpolates two planes (as opposed to frames), the path can be chosen with zero whip spin.

### 7.4 Second extreme: Pure whip spin

Now consider the opposite extreme: Motion takes place entirely within the plane spanned by  $F$ . The viewer is getting to see the same plane rotated within itself. That is, the plane is at rest although the frame may move. This is characterized by the equivalent conditions

$$P' = 0 , \quad \text{and} \quad P F' = F' .$$

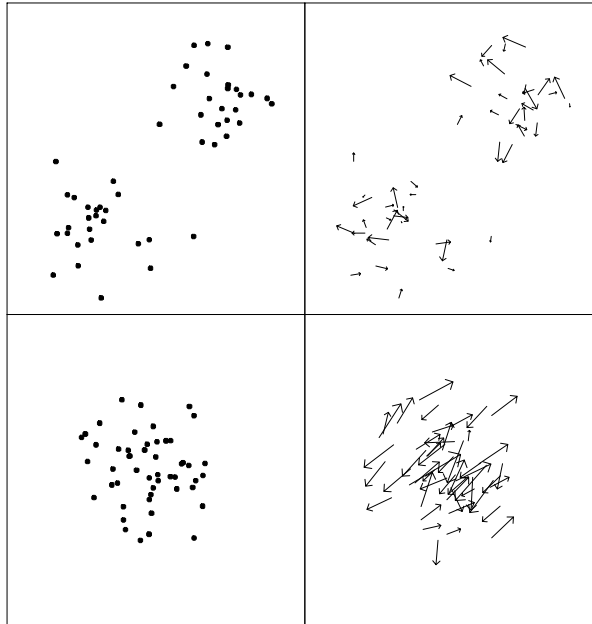


Figure 8: *What we see when we watch moving 2-D projections: A static rendition in terms of scatterplots. The two rows show two projections of a 4-D data set consisting of two well-separated normal clusters. The left column shows location, the right column shows in addition speed represented by arrows. In the top row, the cluster separation is seen in location but not in speed. In the bottom row, the clusters overlap completely in location but are separated in speed: Some arrows point to the upper right, others to the lower left, indicating the presence of two groups.*

Equivalence follows from skew-symmetry of the whip spin matrix  $F^T F'$ .

Pure whip spin is easily generated by superposing a fixed frame  $F_a$  with a dynamic  $d \times d$ -rotation  $V(t)$ :

$$F(t) = F_a V(t) .$$

Obviously, the projection  $P = F F^T = F_a F_a^T$  is fixed, and the whip spin matrix is  $F^T F' = V^T V'$ .

If a tour consists of paths of pure whip spin, the point is to get to see different orientations of  $\text{span}(F_a)$ . This is the case in particular for full-dimensional tours  $d = p$ , see Wegman (1991), Wegman and Luo (1996), and Carr, Wegman and Luo (1996). The full-dimensional case is really the generic case: If  $d < p$  but the plane is at rest, consider this fixed plane as data space, hence  $p = d$  by restriction.

## 8 What We See When We Watch Moving Projections

If a data vector  $\mathbf{x}_i$  is projected onto a moving frame  $F(t)$ , we see some rendering of

$$\text{View}_i(t) = F(t)^T \mathbf{x}_i .$$

In addition, we see the projection of  $\mathbf{x}_i$  in motion, that is, we see where the projected point is going, at least infinitesimally. For a feeble attempt at depicting this fact in a static image, see the right hand plots in Figure 8. In mathematical terms we can say we “see” the derivative

$$View'_i(t) = F'(t)^T \mathbf{x}_i .$$

These two items,  $View_i(t)$  and  $View'_i(t)$ , represent location and velocity of a projected data vector. We can now compare the information they contain:

*Both location and velocity are projections of the data vector  $\mathbf{x}_i$ , the first onto the frame  $F(t)$ , the second onto the frame  $F'(t)$ .*

In contrast to  $F(t)$ , the derivative frame  $F'(t)$  is not usually an orthonormal frame. Nevertheless,  $F'(t)\mathbf{x}_i$  can be interpreted as an oblique projection.

For example, in the common situation of 2-dimensional data projections, velocity adds in principle another 2 dimensions of information to the projection. Thus, when watching a grand tour of 2-dimensional projections in  $p \geq 4$ -dimensional space, we “see” the 4-dimensional subspace  $\text{span}(F(t), F'(t))$ ! If data space is  $p=3$ -dimensional, the additional information provided by velocity is confined to one additional dimension. This generalizes to higher dimensions: The velocity frame  $F'$  can be rank-deficient; in particular when  $d > p/2$ , its rank must be less than that of  $F$ .

We can ask how the location and velocity projections relate to each other. To this end, we use the notion of whip spin. We consider the two extremes of 0% whip spin and 100% whip spin:

- If the frame motion has 0% whip spin, the location frame  $F(t)$  and the velocity frame  $F'(t)$  produce projections onto orthogonal subspaces. This follows because absence of whip spin,  $F^T F' = 0$ , is equivalent to orthogonality of  $\text{span}(F)$  and  $\text{span}(F')$ . The combined dimensionality “seen” in the dynamic projection is  $d + \text{rank}(F')$ , which can be as much as  $2 \cdot d$ .
- If the frame motion has 100% whip spin, then the velocity frame produces a projection onto a subspace of the location subspace:  $\text{span}(F') \subset \text{span}(F)$ . The velocity frame  $F'$  therefore presents all or part of the information contained in the location frame  $F$ , although in a different orientation, and possibly affinely distorted. In a full-dimensional tour with  $d = p$ , the benefit of motion is that two different frames  $F$  and  $F'$  are visible at the same time.

A caveat: We do, of course, not argue that the quality of the information contained in location and velocity are perceptually equivalent. In fact, for the usual rendering methods such as scatterplots, location is vastly superior to velocity. Yet, velocity is not useless: It adds glimpses of additional dimensions that provide a much more holistic view of the geometry of a pointcloud in high-dimensional data space. — As an illustration, we attempt to depict a moving scatterplot of clustered data in Figure 8.

The point of the figure is to show the cluster structure coded in location but not velocity (top row), and coded in velocity but not location (bottom row), respectively. In practice, the coding mode changes from one moment to the next: What is coded as location now may be coded as speed later, and vice versa.

## 9 Kinematics II: Speed of Moving Frames and Planes

### 9.1 Quadratic speed measures

In implementations of dynamic projections, it is desirable to provide viewers with a sense of constancy of motion. This is not just a psychological need but a consideration of practical usefulness: When watching a tour, a viewer is often able to judge the strength of structure by how often it becomes visible. If tour motion is not steady, such judgement is impossible or biased. Erratic motion that changes unpredictably between slow and fast perceived speed is therefore not only disconcerting but potentially misleading.

For tour implementations that are based on interpolation, it is desirable not only to hold speed steady on each leg, but to stitch interpolating paths together in such a way that speed is perceived as about the same on each leg.

The question of how to measure speed does not have a unique answer, even after making strong invariance assumptions and limiting the choices to algebraically simple forms of speed measures.

Reasonable measures of speed are matrix (semi-)norms of the frame derivative  $F'$ . In what follows we consider only speed measures that are derived from quadratic forms of  $F'$ :

$$g_F(F') = \sum_{ijkl} g_{ijkl}(F) F'_{ij} F'_{kl} ,$$

where the coefficients  $g_{ijkl} = g_{ijkl}(F)$  may depend on the current frame  $F$ . Essentially, the derivative  $F'$  is strung out as a  $p \cdot d$ -dimensional vector, and  $g_F$  is a quadratic function thereof. The necessity of allowing  $g = g_F$  to depend on the current frame  $F$  stems from the fact that the space of derivatives  $F'$  differs for different  $F$ 's.

Assuming  $g_F$  is non-negative, speed at  $F$  is measured by  $g_F(F')^{1/2}$ . In order to simplify language, we use the term ‘‘speed measure’’ for both  $g_F(F')^{1/2}$  and  $g_F(F')$ , although only  $g_F^{1/2}$  is speed in the proper sense.

The simplest example of a quadratic speed measure is the Euclidean squared norm of  $F'$  considered as a  $p \cdot d$ -vector, which is also called the squared Frobenius norm of  $F'$ :

$$g_F(F') = \|F'\|_{Frob}^2 = \sum_{ij} F'^2_{ij} = \text{trace}(F'^T F') = \text{trace}(F' F'^T) .$$

This example is intuitive as we will see below, but it is atypical because the coefficients do not depend on  $F$ :  $g_{ijkl} = 0$  for  $(ij) \neq (kl)$  and  $g_{ijij} = 1$ .



## 9.2 Reducing the set of speed measures: Invariance requirements

We need to develop criteria to whittle down the universe of speed measures to an intuitively plausible subset. Criteria that are particularly intuitive and mathematically powerful are invariance requirements under orthogonal transformations of both

- the  $p$ -dimensional data space and
- the  $d$ -dimensional projection plane.

Requiring invariance is a normative act; it is not something that can be derived from still higher principles. One can, however, make informal arguments: Invariance is desirable because the notion of speed should be devised a priori without looking at the structure of particular data. A priori we do not have knowledge of interesting projections of data space, that is, projections near which it might be worthwhile to slow down and linger. Neither do we have a priori knowledge about how to orient the projection plane to our advantage. The purpose of the two invariance requirements is to level the playing field among dynamic projections and their orientations. While it is true that potentially interesting data projections can be found with multivariate analysis and projection pursuit, this fact is quite irrelevant when trying to gauge speed measures of data projections; the only data that can possibly serve as a gauge are structureless or null data. A suitable set of null data is formed by data that “look the same from all sides,” or equivalently, that are rotationally symmetric, such as multivariate standard normal data, which incidentally serve as a null gauge for exploratory projection pursuit as well (Friedman 1987). We will make use of null data in Section 9.4.

Underlying any invariance requirement with regard to orthogonal transformations is of course a Euclidean metric in data space. We have made use of Euclidean metrics from the moment when we required projection frames to be orthonormal and projections to be orthogonal. The necessity of Euclidean metrics is unquestioned, although no single best metric may exist in any particular data analysis.

## 9.3 The structure of invariant speed measures

We proceed with the mathematical definition of invariance. Denote by  $O(p)$  the group of orthogonal transformations  $U$  ( $U^T U = I_p$ ) of  $p$ -dimensional data space. Orthogonal transformations comprise proper rotations (determinant +1), reflections, and their compositions. Similarly, denote by  $O(d)$  the group of orthogonal transformations  $V$  ( $V^T V = I_d$ ) of the  $d$ -dimensional projection plane.

**Definitions:** A speed measure  $g_F(F')$  is

- **left-invariant** if two paths  $F(t)$  and  $UF(t)$  have the same speeds:  $g_{UF}(UF') = g_F(F')$  for all  $U \in O(p)$  and all  $t$ ;
- **right-invariant** if two paths  $F(t)$  and  $F(t)V$  have the same speeds:  $g_{FV}(F'V) = g_F(F')$  for all  $V \in O(d)$  and all  $t$ .

Note that the transformations  $U$  and  $V$  are fixed: They do *not* vary with time  $t$ .

The speed measure based on the Frobenius norm,  $g_F(F') = \|F'\|_{Frob}^2$ , is both left-invariant and right-invariant, as is easily seen with basic trace manipulations:  $\text{trace}((UF'V)^T(UF'V)) = \text{trace}(F'^T F')$ .

In preparation of the main result of this subsection, we introduce some definitions: For any speed measure  $g_F$  there are two associated component speed measures derived from the decomposition of frame motion  $F'$  into whip spin  $PF'$  and plane motion  $(I - P)F'$ :

1. *Frame speed*:  $g_F(F')$ ,
2. *Whip speed*:  $g_F(PF')$ ,
3. *Plane speed*:  $g_F((I - P)F')$ ,

where each is really squared speed. The following theorem states two consequences of invariance: 1) Frame speed has a Pythagorean decomposition into whip speed and plane speed, and 2) whip speed and plane speed are just Frobenius norms up to a scalar factor:

**Theorem 2:** *Squared speed measures  $g_F(F')$  that are both left-invariant and right-invariant are of the form*

$$\begin{aligned} g_F(F') &= \alpha_w \cdot \|PF'\|_{Frob}^2 + \alpha_p \cdot \|(I - P)F'\|_{Frob}^2 \\ &= \alpha_w \cdot \|F'^T F'\|_{Frob}^2 + \alpha_p/2 \cdot \|P'\|_{Frob}^2 \end{aligned}$$

for some  $0 \leq \alpha_w, \alpha_p \leq 1$ .

The proof is in the Appendix (Sections 12.2 and 12.3).

The first equality is the main result. The second equality is an algebraic reformulation with intuitive appeal: Whip speed is just a function of the whip spin matrix, and plane speed is just a function of the path of projections  $P = P(t)$ .

With theorem 2 we are down to speed measures that are not too different from the basic Frobenius norm: For  $\alpha_w = \alpha_p = 1$ , we have  $g_F(F') = \|F'\|_{Frob}^2$ . Because speed measures are obviously equivalent if they differ only by a scale factor, the remaining freedom is that of choosing the relative weighting of whip speed and plane speed, making this essentially a one-parameter family of speed measures.

Before we go into specific choices, here is a short discussion of cases where the choice is irrelevant:

- **Pure plane motion:** These are the preferred paths for tours that use low-dimensional planes, typically  $d \leq 3$ , with rendering methods for which orientation is unimportant. For such paths there is essentially only one way to measure speed, modulo an irrelevant choice  $\alpha_p = 1$ . Because  $PF' = 0$ , we have  $(I - P)F' = F'$ , hence

$$g_F(F') = \|F'\|_{Frob}^2 = 1/2 \cdot \|P'\|_{Frob}^2,$$

up to an irrelevant choice  $\alpha_p = 1$ . This is only a function of the underlying plane path  $P$ , as it should be.

- **Pure whip spin:** These are the paths used when the plane is at rest as in full-dimensional tours with  $d = p$ . Again, there is essentially only one way to measure speed: Because  $(I - P)F' = 0$ , we have

$$g_F(F') = \|F'\|_{Frob}^2 = \|F'^T F'\|_{Frob}^2 ,$$

up to an irrelevant choice  $\alpha_w = 1$ . In particular this implies that for paths of  $p \times p$  frames  $U(t) \in O(p)$  the invariant speed measure is essentially unique:  $g_U(U') = \|U'\|_{Frob}^2$ .

#### 9.4 Choosing among speed measures that are left- and right-invariant

The relative weighting of whip speed and plane speed is not something that can be decided on mathematical grounds as strong as left- and right-invariance. We know of auxiliary arguments in favor of two particular choices:

- $\alpha_w = 1, \alpha_p = 1$ ,
- $\alpha_w = 1, \alpha_p = 2$  .

**Average squared speed of projected null data:** For the speed measure defined by  $\alpha_w = 1, \alpha_p = 1$  ( $\|F'\|_{Frob}^2$ ) there exists an interpretation in terms of average squared speed of null data as seen in the projection. Null data is data with no or little structure in a sense to be specified. It is intuitive to choose null data to gauge overall speed of a moving point cloud on a computer screen: If speed — as quantified by a speed measure — is held constant, the viewer should perceive null data as moving at constant overall speed. In practice this can be used as follows: When viewing real data, perceived variations in overall speed act as auxiliary indicators for the presence of structure.

In order to specify the exact meaning of “null data,” it is convenient to use random vectors rather than finite data. Let  $\mathbf{X}$  be a  $p$ -dimensional random vector with existing second moments ( $E[\|\mathbf{X}\|^2] < \infty$ ,  $E$  denotes expectation). Null data for our purposes are formed by a random vector with spherically symmetric second moments:  $E(\mathbf{X}\mathbf{X}^T) = \alpha \cdot I_p$ . That is, this data “looks the same” from all sides as far as second moments are concerned.

The projection of  $\mathbf{X}$  onto a moving  $d$ -frame  $F = F(t)$  is the time-dependent  $d$ -dimensional random vector  $F^T \mathbf{X}$ , with velocity  $F'^T \mathbf{X}$  (another time-dependent random vector) and squared speed  $\|F'^T \mathbf{X}\|^2$  (a time-dependent random variable; note this is the Euclidean norm in  $\mathbb{R}^d$ ).

**Fact:** *If the second moment matrix is rotationally symmetric,  $E(\mathbf{X}\mathbf{X}^T) = \alpha \cdot I_p$ , then the average squared speed of the projected data is*

$$E[\|F'^T \mathbf{X}\|^2] = \alpha \cdot \|F'\|_{Frob}^2 .$$

The proof is elementary, observing that  $\|F'^T \mathbf{X}\|^2 = \text{trace}(F'F'^T \mathbf{X}\mathbf{X}^T)$ .

An example of second-order null data is the  $p$ -dimensional standard normal, but so is any spherically symmetric distribution in  $\mathbb{R}^p$ . Because we do not require centered variables, “data” consisting of the variable unit vectors form another example: Let  $\mathbf{X}$  be a random vector that takes on each of the  $p$  variable unit vectors with probability  $1/p$ ; the second moment matrix is  $1/p \cdot I_p$ . Thus,  $\|F'\|_{Frob}^2$  can be interpreted as essentially the average squared speed of the projected variable unit vectors.

**Average squared speed of rotating null data:** For the speed measure defined by  $\alpha_w = 1$ ,  $\alpha_p = 2$ , there exists an interpretation in terms of the intuitive notion of “data rotation.” We implicitly used this speed measure in Asimov and Buja (1994) because it is mathematically the simplest choice (see Section 10).

The term “data rotation” suggests rotating data space rather than the projection plane. Hence we fix the projection frame once for all, for example, at the frame  $E_d$  consisting of the first  $d$  variable unit vectors. Let  $U = U(t) \in SO(p)$  be a path of rotations of  $p$ -dimensional data space ( $SO(p)$  is the group of orthogonal maps on  $\mathbb{R}^p$  with determinant  $+1$ ).

We rotate the data with the inverse of  $U$ , that is, a data vector  $\mathbf{x}_i$  gets mapped to  $U(t)^T \mathbf{x}_i$ . We then project it onto the first  $d$  variable directions:  $View_i(t) = E_d^T U(t)^T \mathbf{x}_i$ . In effect, this amounts to a projection onto the frame  $F(t) = U(t)E_d$ , that is, the first  $d$  columns of  $U$ . We write  $U(t) = (F(t), G(t))$ , where  $G$  is the complement frame of size  $p \times (p - d)$ . The velocity vector of the rotating data vector  $U(t)^T \mathbf{x}_i$  in  $p$ -space is  $U'(t)^T \mathbf{x}_i$ , and the squared speed is  $\|U'(t)^T \mathbf{x}_i\|^2$ . Describing null data again by a random vector  $\mathbf{X}$  with rotation symmetric second moments  $E[\mathbf{X}\mathbf{X}^T] = \alpha \cdot I_p$ , the average squared speed of the rotating data  $U^T \mathbf{X}$  is

$$E[\|U'(t)^T \mathbf{X}\|^2] = \alpha \cdot \|U'(t)\|_{Frob}^2,$$

which makes sense because  $\|U'\|_{Frob}$  is essentially the only way to measure speed of “full-dimensional frames”  $U(t)$ .

We now examine  $\|U'(t)\|_{Frob}$  as a potential speed measure for the moving frame  $F(t) = U(t)E_d$ . We start with the following rationale: When measuring speed of a data rotation  $U = (F, G)$  as visible in the projection onto  $F$ , it should be required that the invisible whip spin in the orthogonal complement does not contribute. Therefore, we ask that the complement frame  $G$  have zero whip spin, which is possible by theorem 1. The result is:

**Fact:** *If a path of  $d$ -frames  $F(t)$  is augmented with a path of  $(p - d)$ -frames  $G(t)$  that have zero-whip spin to form a path of data rotations  $U(t) = (F(t), G(t))$ , then the speed of the data rotation is measured by*

$$\|U'\|_{Frob}^2 = \|PF'\|_{Frob}^2 + 2 \cdot \|(I - P)F'\|_{Frob}^2.$$

The proof is in Section 12.9. As a result, we have an interpretation of the speed measure corresponding to  $\alpha_w = 1$  and  $\alpha_p = 2$  in terms of speed of most restful rotations of null data with regard to the path of frames  $F(t)$ .

## 10 Optimal Paths of Frames

Metrics on paths of frames do not only allow us to measure speed but to find shortest paths between two given frames. The interpretation for data visualization is that such paths carry one data projection onto another data projection in the most restful manner.

As we have learned, there exists a multiplicity of metrics that are reasonable. This poses a quandary: Each metric has its own shortest paths. In two specific situations, however, the quandary does not exist: 1) when using paths of frames that consist of pure plane motion, as in tours of low-dimensional projections where orientation is irrelevant; and 2) when using paths of frames that consist of pure whip spin, as in full-dimensional tours. This is not too surprising because we found that both pure plane motion and pure whip spin are measured in essentially unique ways, the choice arising from combining the two speed components into an overall speed measure.

In what follows, we replace the notion of “shortest path” with that of “geodesic path,” which means “locally shortest.” This notion is more useful and more convenient on a curved manifold such as the one formed by  $d$ -frames in  $p$ -space.

In order to introduce geodesic paths of frames, we use a simplification:

**Convention:** *The starting frame  $F_a$  is the unit frame  $E_d$ .*

The case of a general starting frame  $F_a$  is obtained by mapping  $E_d$  with some  $U_a \in SO(p)$  to  $F_a$ , so that paths  $F(t)$  through  $E_d$  map to paths  $U_a F(t)$  through  $F_a$ . Due to left-invariance of speed measures,  $F(t)$  and  $U_a F(t)$  have the same speed properties, which implies that a geodesic through  $E_d$  will be mapped to a geodesic through  $F_a$ .

In order to describe geodesic paths of frames, we need some facts about the connection between rotations and their infinitesimal generators via matrix exponentials  $\exp(S) = \sum_0^\infty S^n/n!$ :

### Facts:

- *Any skew matrix  $S$  of size  $p \times p$  can be block-diagonalized with skew blocks of size  $2 \times 2$  in a suitable orthonormal coordinate system.*
- *Any rotation  $U \in SO(p)$  can be block-diagonalized with blocks consisting of  $2 \times 2$  rotations (and a trailing  $+1$  if  $p$  is odd) in a suitable orthonormal coordinate system.*
- $\exp \begin{pmatrix} 0 & -t \\ t & 0 \end{pmatrix} = \begin{pmatrix} c_t & -s_t \\ s_t & c_t \end{pmatrix}$
- *For all  $p \times p$  skew matrices  $S$ :  $\exp(S) \in SO(p)$ .*

The first two facts show how the third fact implies the fourth, all of which is a consequence of the spectral theorem for unitary transformations and its translation to orthogonal transformations (Halmos 1958, Section 81 and 82).

Exponentials of skew matrices can be used to generate paths of rotations  $U(t)$  by spinning at constant but different speeds in sets of mutually orthogonal 2-planes. This is expressed by  $U(t) = \exp(St)$ . With this background, we can formulate a class of paths of frames that contain all geodesics for all left- and right-invariant metrics:

$$(*) \quad F(t) = \exp(St) E_d \exp(Qt), \quad S \text{ is } p \times p, Q \text{ is } d \times d, \text{ both skew.}$$

The  $p \times p$  path  $\exp(St)$  transports the starting frame  $E_d$  through  $p$ -space, while the  $d \times d$  path  $\exp(Qt)$  applies an additional whip spin to  $\exp(St) E_d$ . This class of paths of frames is quite rich. In order to sort out the members that are geodesic w.r.t. a given metric, we need notation for the canonical blocks of  $S$ :

$$S = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix} \quad \begin{array}{l} \} d \\ \} (p-d) \end{array}$$

**Theorem 3:** *The geodesic paths of frames with regard to the metric given by  $\alpha_w$  and  $\alpha_p$  are exactly the paths of the form (\*) satisfying  $Q = (\frac{\alpha_p}{2\alpha_w} - 1)S_{11}$  and  $S_{22} = 0$ .*

The proof is in the Appendix (Sections 12.5 and 12.6).

This theorem is surprisingly hard to apply when the goal is to interpolate  $F_a = E_d$  and an arbitrary target frame  $F_z$ . Note that for a given target  $F_z$  the problem is to construct a suitable  $S$  and its associated  $Q$  such that  $F_z = \exp(S) E_d \exp(Q)$ . In Asimov and Buja (1994) we solved the problem for 2-frames with regard to the metric defined by  $\alpha_w = 1$  and  $\alpha_p = 2$ , which is surprisingly the simplest case because  $Q$  magically vanishes:  $Q = (\frac{\alpha_p}{2\alpha_w} - 1)S_{11} = 0$ . The construction of  $S$  from  $F_z$  is still an open problem for all other metrics, including the straight Frobenius metric defined by  $\alpha_w = \alpha_p = 1$ .

The situation is not so bleak, though, because two cases of particular interest are readily solved: 1) pure plane motion for low-dimensional tours where orientation is irrelevant, and 2) pure whip spin for full-dimensional tours.

**Corollary:** *Paths of theorem 3 are universally geodesic for all left- and right-invariant metrics in the following cases:*

- $S_{11} = 0$ , that is, pure plane motion;
- $S_{12} = 0$ , that is, pure whip spin.

*In the first case  $Q = 0$  by necessity, and in the second case  $Q$  can be absorbed into  $S_{11}$ , hence  $Q = 0$  w.l.o.g.; thus  $F(t) = \exp(St) E_d$ . The matrix  $S$  is of the following form, respectively:*

$$S = \begin{pmatrix} 0 & S_{12} \\ S_{21} & 0 \end{pmatrix}, \quad S = \begin{pmatrix} S_{11} & 0 \\ 0 & 0 \end{pmatrix}.$$

The first case is steady rotation vertically out of the current plane, and the second case is equally steady rotation within the current plane. In order to fully appreciate this interpretation, it helps to cast  $S$  and  $\exp(St)$  in canonical form as mentioned in the list of facts at the beginning of the section. Here are two examples:

- Pure plane motion of a 2-frame in 4-space ( $d = 2, p = 4$ , canonical coordinates): The 1- and 2-axis in the starting plane are being rotated towards the 3- and 4-axis at speeds  $u$  and  $v$ , respectively. The (1,3)-plane and the (2,4)-plane are invariant.

$$S = \begin{pmatrix} 0 & 0 & -u & 0 \\ 0 & 0 & 0 & -v \\ u & 0 & 0 & 0 \\ 0 & v & 0 & 0 \end{pmatrix} : \quad \exp(St) = \begin{pmatrix} c_{ut} & 0 & -s_{ut} & 0 \\ 0 & c_{vt} & 0 & -s_{vt} \\ s_{ut} & 0 & c_{ut} & 0 \\ 0 & s_{vt} & 0 & c_{vt} \end{pmatrix},$$

- Pure whip spin of a 4-frame in 4-space ( $d = p = 4$ , canonical coordinates): The (1,2)-plane and the (3,4)-plane are invariant in the full 4-space. The rotation speeds are  $u$  and  $v$ , respectively.

$$S = \begin{pmatrix} 0 & -u & 0 & 0 \\ u & 0 & 0 & 0 \\ 0 & 0 & 0 & -v \\ 0 & 0 & v & 0 \end{pmatrix} : \quad \exp(St) = \begin{pmatrix} c_{ut} & -s_{ut} & 0 & 0 \\ s_{ut} & c_{ut} & 0 & 0 \\ 0 & 0 & c_{vt} & -s_{vt} \\ 0 & 0 & s_{vt} & c_{vt} \end{pmatrix}.$$

Note that for  $d = p$ , the condition  $S_{12} = 0$  is trivially satisfied because  $S_{12}$  is of size  $d \times (p - d) = d \times 0$ .

In computer implementations of geodesic interpolation, the problem is in either case to find the canonical coordinates in which a path can be expressed as simply as above. Details are carried out in the companion paper by Buja et al. (2004).

We can gain some intuition into the structure of geodesic paths by visualizing a moving frame with the same means that they are intended to serve: projections from high-dimensional space. Consider a path of 2-frames in 4-space,

$$F(t) = \exp(St)E_d = ((c_{ut}, 0, s_{ut}, 0)^T, (0, c_{vt}, 0, s_{vt})^T),$$

generated by the above example of pure plane motion. Interpret  $F(t)$  as a curve in  $\mathbb{R}^8$ , eliminate the trivial zeros to make this a curve in  $\mathbb{R}^4$  and form 2-D projections of the curve. A random projection from a grand tour in XGobi is shown in Figure 9. The particular curve has  $u = .1$  and  $v = 2^{1/2}$ . It should not be too surprising that it seems to be filling up a torus-like surface. The underlying mathematical reason is that the mapping  $(u, v) \mapsto (c_u, s_u, c_v, s_v)$  is a parametrization of a 2-D torus in 4-space, and  $t \mapsto (c_{ut}, s_{ut}, c_{vt}, s_{vt})$  is a dense curve on the torus iff  $u/v$  is irrational.

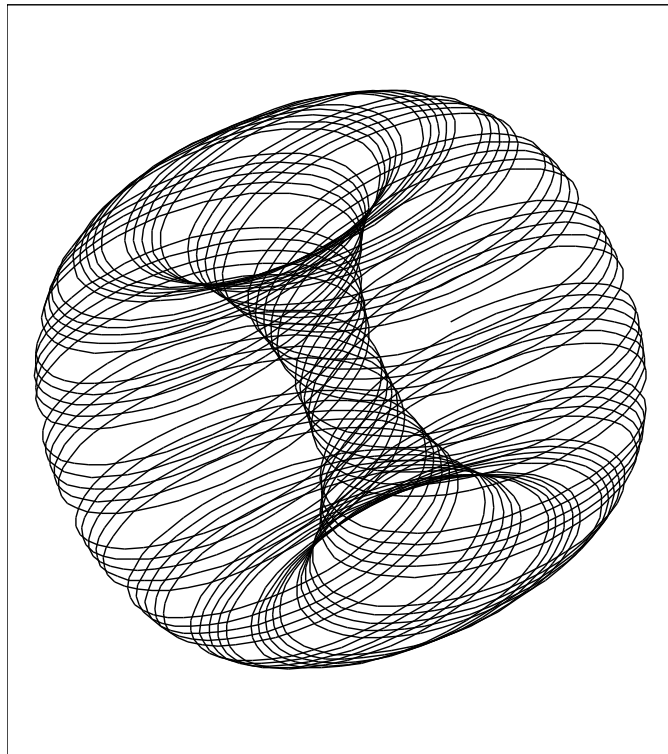


Figure 9: *A random 2-projection of a geodesic path of 2-frames in 4-space, interpreted as a curve in 8-space (or 4-space after eliminating four zeros). The figure shows a finite piece of a single non-intersecting path filling up a surface that is topologically a 2-D torus.*

## 11 Conclusions

Dynamic projections form a powerful set of tools for viewing high-dimensional data and mathematically defined high-dimensional objects. Dynamic projections exploit the human eye's natural ability to detect and recognize objects in motion. In addition, dynamic projections exploit the human instinct for playfulness. For graphical data analysis, we can turn a humble static 2-D projection into a rocking image that conveys up to four simultaneous dimensions at any given time.

The goal of this paper was to give a mathematical framework for dynamic projections: We discussed graphical rendering of projections of data space onto planes of dimensions 1, 2, 3 and  $> 3$ . We analyzed the role of orientation of projections. We proposed notions that allow us to decompose motion along paths of projections into two natural components: within-plane spin and plane motion. Pure plane motion is arguably optimal for most types of low-dimensional rendering, but for full-dimensional motion there is no plane motion and all that matters are dynamic changes in orientation. We discussed ways of measuring speed of moving projections and found a wealth of mathematical structure that lead to notions of geodesic motion of projections.



## 12 Appendix: Invariant Metrics for Frames and Planes

The goal of this appendix is to describe invariant Riemannian metrics on the Stiefel manifold  $V_{d,p}$  of orthonormal  $d$ -frames in  $p$ -space. The purpose of Riemannian metrics is to measure speed of moving frames. There are two types of invariance: Left-invariance requires that for fixed  $U \in O(p)$ , the paths  $F(t)$  and  $UF(t)$  show the same speed at each time point  $t$ . Right-invariance requires that for fixed  $V \in O(d)$ , the paths  $F(t)$  and  $F(t)V$  show the same speed at each time point  $t$ .

Left-invariance implies a structure theorem. Part of the theorem says that the two components of frame motion — pure whip spin and pure plane motion — are always orthogonal, implying a Pythagorean relation among the two.

If we assume right invariance in addition to left-invariance, then whip speed and plane speed are each essentially measured by Frobenius norms. Nothing is implied, however, about the relative weighting of the two types of speed when merging them into an overall speed measure.

Finally, we characterize Riemannian metrics on the Grassmann manifold  $G_{d,p}$  of  $d$ -planes in  $p$ -space, but we think of these metrics as semi-metrics on  $V_{d,p}$  that are invariant under dynamic within-plane spin. More precisely, “Grassmann semi-metrics” are required to assign the same speed properties to a path of frames  $F(t)$  and any other path  $F(t)V(t)$  that differs from the first only by a dynamic within-plane rotation  $V(t)$ . Such Grassmann semi-metrics are essentially unique.

Below, we introduce the concepts of tangent space and Riemannian metric of differential geometry, but we immediately apply them to the Stiefel manifold. The presentation is self-contained, and no knowledge of differential geometry is assumed. The material belongs really in the framework of reductive homogeneous spaces (Kobayashi and Nomizu 1969, chap. X), but we found it possible to give an elementary introduction.

A word to the mathematician: Because we adopt invariance with regard to  $O(p)$  rather than  $SO(p)$ , we rule out the curious appearance of non-equivalent invariant metrics on the Grassmann manifold  $G_{2,4}$ . It seems to us that these metrics are irrelevant for visualization with dynamic projections. Similarly “unnatural” metrics exist on  $S_{d,p}$  when  $p - d = 2$ , and they are equally ruled out.

### 12.1 Metrics for frames

**Definition:** *The tangent space  $T_F$  at the frame  $F \in V_{d,p}$  is the space of all  $p \times d$  matrices  $X$  that are derivatives of differentiable paths of frames through  $F$ :*

$$T_F = \{X \mid \text{there exists a path } F(t) \text{ such that } F(0) = F, F'(0) = X \}$$

**Fact:**  $T_F = \{X \mid (F^T X)^T = -(F^T X) \}$

**Proof:** The inclusion  $\subset$  is easily shown by taking derivatives of the orthonormality condition  $F(t)^T F(t) = I_d$ , which for each  $t$  represents the set of equations that define

$V_{d,p}$ . It remains to show that every such  $X$  is a derivative matrix. By a change of coordinates, we can achieve that  $F$  consists of the first  $d$  columns of the identity matrix  $I_p$ . The matrix  $F^T X$  is then the top  $d \times d$  block of  $X$ , hence the condition  $(F^T X)^T = -(F^T X)$  says that this block is skew. Therefore  $X$  can be extended to a skew matrix  $S$  of size  $p \times p$ . (This extension is not unique.) The matrix exponential  $U(t) = \exp(St)$  is a path of orthogonal matrices that satisfy  $U(0) = I_p$  and  $U'(0) = S$ . Therefore, the first  $d$  columns form a path of frames  $F(t)$  with  $F(0) = F$  and  $F'(0) = X$ .  $\square$

The matrix  $F^T X$  is what we called the whip spin matrix. The above condition says that every  $p \times d$  matrix with skew whip spin matrix with regard to  $F$  is an element of  $T_F$ . Skew symmetry of a  $d \times d$  matrix implies  $(d+1)d/2$  independent equality constraints, hence:

**Fact:**  $\dim(V_{d,p}) = \dim(T_F) = pd - (d+1)d/2$

**Definition:** A **Riemannian metric** is a family of symmetric bilinear forms  $g_F(.,.)$  defined on  $T_F$  (and differentiable in  $F$  in a suitable sense).

Riemannian metrics can be thought of as “infinitesimal inner products.” The usual definitions of Riemannian metrics include positive definiteness, but we will also consider degenerate bilinear forms, that is, semi-metrics.

Symmetric bilinear forms are uniquely determined by their quadratic forms which, by abuse of notation, we also denote by  $g_F$ :

$$g_F(X) = g_F(X, X) , \quad g_F(X, Y) = (g_F(X + Y) - g_F(X - Y))/4 .$$

We therefore use the term “Riemannian metric” both for  $g_F(.,.)$  and  $g_F(.)$ .

## 12.2 Left-invariant metrics for frames

We turn to left-invariance of Riemannian metrics with regard to  $O(p)$ , the group of orthogonal  $p \times p$  matrices. We use invariance with regard to  $O(p)$  as opposed to  $SO(p)$ , the orthogonal transformations with determinant  $+1$ . The difference is that  $O(p)$  includes reflections besides proper rotations. Although  $SO(p)$ -invariance is standard in differential geometry, we decided in favor of  $O(p)$ -invariance for two reasons: 1) We see no reason why a path of frames and a reflected version thereof should have different speed properties. 2) Some unintuitive technical artifacts can be avoided, such as strange invariant metrics on the Grassmannian  $G_{2,4}$ , peculiar to these dimensions and unrelated to any visualization properties of dynamic projections.

For any  $U \in O(p)$ , if  $F(t)$  is a path of frames, so is  $UF(t)$ , hence:

**Fact:**  $U \in O(p) , \quad X \in T_F \quad \implies \quad UX \in T_{UF} .$

**Definition:** A Riemannian metric is left-invariant w.r.t.  $O(p)$  if it satisfies

$$g_{UF}(UX, UY) = g_F(X, Y) \quad \text{or, equivalently,} \quad g_{UF}(UX) = g_F(X),$$

for all  $U \in O(p)$ .

**Theorem:** A left-invariant Riemannian metric satisfies

$$g_F(UX) = g_F(X) \quad \text{for all } U \text{ such that } UF = F.$$

Conversely, any positive definite quadratic form  $g(X)$  defined on a single tangent space  $T_F$  satisfying

$$(*) \quad g(UX) = g(X) \quad \text{for all } U \text{ such that } UF = F$$

can be extended to a unique left-invariant Riemannian metric through

$$g_{UF}(UX) \stackrel{\text{def}}{=} g(X) \quad \text{for all } U \in O(p).$$

**Proof:** The first part is a trivial specialization of the definition of left-invariance.

For the second part, assume we have two representations of a frame  $U_1F = U_2F$  and two representations of a tangent vector  $U_1X_1 = U_2X_2 \in T_{U_1F}$  where  $X_1, X_2 \in T_F$ . We have to show  $g_{U_1F}(U_1X_1) = g_{U_2F}(U_2X_2)$  in order to make the above definition consistent.

The defining equations are  $g_{U_1F}(U_1X_1) = g(X_1)$  and  $g_{U_2F}(U_2X_2) = g(X_2)$ , hence we have to show  $g(X_1) = g(X_2)$ .

From  $U_1X_1 = U_2X_2$  we get  $X_2 = U_2^T U_1 X_1$ , and from  $U_1F = U_2F$  we get  $U_2^T U_1 F = F$ . Hence, we can apply the assumption (\*) with  $U = U_2^T U_1$  and get  $g(X_2) = g(UX_1) = g(X_1)$ .  $\square$

### Reduction of the problem:

Given the above theorem, we can analyze left-invariant Riemannian metrics by choosing a convenient frame  $F$  and by analyzing the structure of symmetric bilinear and quadratic forms  $g$  on  $T_F$  satisfying property (\*).

The most convenient frame is  $E_d = (e_1, \dots, e_d)$ , that is, the frame consisting of the first  $d$  columns of the identity matrix  $I_p$ . From now on we write  $g = g_{E_d}$ , and we assume that  $g$  satisfies condition (\*).

We need to characterize the tangent space  $T_{E_d}$  and specialize property (\*). To this end, for a  $p \times d$  matrix  $X$  and for  $U \in O(p)$  we write

$$X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}, \quad U = \begin{pmatrix} U_{11} & U_{12} \\ U_{21} & U_{22} \end{pmatrix},$$

where  $X_1$  is  $d \times d$ ,  $X_2$  is  $(p-d) \times d$ ,  $U_{11}$  is  $d \times d$ ,  $U_{12}$  is  $d \times (p-d)$ , and so on.

**Fact:**  $T_{E_d} = \{X \mid X_1 \text{ is skew symmetric} \}$

**Fact:**  $UE_d = E_d \iff U = \begin{pmatrix} I_d & 0 \\ 0 & U_{22} \end{pmatrix}, U_{22} \in O(p-d).$

**Proofs:**  $X_1 = E_d^T X$  is the whip spin matrix. The condition  $UE_d = E_d$  implies  $U_{11} = I_d$  and  $U_{21} = 0$ , and the rest follows from orthogonality of the columns of  $U$ .  $\square$

**Lemma:**  $X = \begin{pmatrix} X_1 \\ 0 \end{pmatrix}$  and  $Y = \begin{pmatrix} 0 \\ Y_2 \end{pmatrix} \implies g(X, Y) = 0.$

That is, whip spin and plane motion are orthogonal under any left-invariant Riemannian metric. As a consequence:

**Corollary:**  $g(X) = g\left(\begin{pmatrix} X_1 \\ 0 \end{pmatrix}\right) + g\left(\begin{pmatrix} 0 \\ X_2 \end{pmatrix}\right).$

**Proof:** We use condition (\*). Let  $R$  be such that  $UE_d = E_d$ , that is  $U_{11} = I_d$ ,  $U_{21} = 0$ ,  $U_{12} = 0$ ,  $U_{22} \in O(p-d)$ . From this and  $X_2 = 0$  follows  $UX = X$ . It also follows that  $UY = \begin{pmatrix} 0 \\ U_{22}Y_2 \end{pmatrix}$  because  $Y_1 = 0$ . We now use the conclusion of (\*) in the form  $g(X, Y) = g(UX, UY)$ :

$$g(X, Y) = g(UX, UY) = g\left(X, \begin{pmatrix} 0 \\ U_{22}Y_2 \end{pmatrix}\right), \text{ for all } U_{22} \in O(p-d).$$

Consider matrices  $Y_2$  in which all but one column is zero: It follows that  $g(X, Y)$  is a  $O(p-d)$ -left-invariant *linear* form of these special matrices. Such left-invariant linear forms are identically zero. The  $d$  spaces of such special matrices span the space of all  $Y_2$  matrices, hence  $g(X, Y) = 0$ , for all  $Y_2$ .  $\square$

**Fact:**  $g\left(\begin{pmatrix} X_1 \\ 0 \end{pmatrix}\right)$  can be any quadratic form of  $X_1$ .

**Lemma:**  $X = \begin{pmatrix} 0 \\ X_2 \end{pmatrix}$  and  $Y = \begin{pmatrix} 0 \\ Y_2 \end{pmatrix} \implies g(X, Y) = \text{trace}(X_2^T Y_2 C)$  for some symmetric  $d \times d$  matrix  $C$ . The metric is positive definite on  $X$ 's of the form  $X = \begin{pmatrix} 0 \\ X_2 \end{pmatrix}$  iff  $C$  is positive definite.

**Proof:** For  $X$  and  $Y$  as above, any bilinear form  $g(X, Y)$  is of the form

$$g(X, Y) = \sum_{i,j=1\dots d} \mathbf{x}_i^T A_{ij} \mathbf{y}_j,$$

where  $X_2 = (\mathbf{x}_1, \dots, \mathbf{x}_d)$ ,  $Y_2 = (\mathbf{y}_1, \dots, \mathbf{y}_d)$  and  $A_{ij}$  are matrices of size  $(p-d) \times (p-d)$ .

We consider special matrices  $X_2$  and  $Y_2$  that have only one non-zero column each, which we denote  $\mathbf{x}_i$  and  $\mathbf{y}_j$ , respectively. We have

$$g(X, Y) = \mathbf{x}_i^T A_{ij} \mathbf{y}_j .$$

With property (\*), we also have for  $UE_d = E_d$ :

$$g(X, Y) = g(UX, UY) = \mathbf{x}_i^T U_{22}^T A_{ij} U_{22} \mathbf{y}_j ,$$

from which follows

$$A_{ij} = U_{22}^T A_{ij} U_{22} , \quad \text{for all } U_{22} \in O(p-d) .$$

By a) of the corollary in Section 12.7, this implies that  $A_{ij}$  is multiple of the identity:

$$A_{ij} = c_{ij} \cdot I_{p-d} .$$

It follows that for general  $X_2$  and  $Y_2$  we have

$$g(X, Y) = \sum_{i,j=1\dots d} \mathbf{x}_i^T A_{ij} \mathbf{y}_j = \sum_{i,j=1\dots d} \mathbf{x}_i^T \mathbf{y}_j c_{ij} = \text{trace}(X_2^T Y_2 C) ,$$

where  $C = (c_{ij})$ . Symmetry of  $C$  follows from symmetry of  $g(\cdot, \cdot)$ .

For positive definiteness, let  $C = \sum_i \lambda_i \mathbf{u}_i \mathbf{u}_i^T$  be an eigendecomposition of  $C$ . We get

$$g(X) = \text{trace}(X_2^T X_2 C) = \sum_i \lambda_i \|X_2 \mathbf{u}_i\|^2 ,$$

where  $\|\cdot\|$  is the Euclidean norm on  $\mathbb{R}^{p-d}$ . Thus  $g(X)$  is positive definite iff all  $\lambda_i$  are positive, that is, if  $C$  is positive definite.  $\square$

The following theorem is a summary and reformulation of the above lemmas for an arbitrary frame  $F$ . Note that  $X_1$  at  $F = E_d$  corresponds to  $F^T X$  at an arbitrary  $F$ . Thus whip speed can be measured by an arbitrary non-negative quadratic function  $g_{whip}(F^T X)$  of the whip spin matrix  $F^T X$ . As for plane motion, the matrix  $X_2^T X_2$  at  $F = E_d$  becomes  $((I - P)X)^T ((I - P)X) = X^T (I - P)X$  at an arbitrary  $F$ . We finally translate  $\text{trace}(X^T (I - P)XC) = \|(I - P)XC^{1/2}\|_{Frob}^2$ .

**Theorem:** *Left-invariant Riemannian metrics are of the form*

$$g_F(X) = g_{whip}(F^T X) + \|(I - P)XC^{1/2}\|_{Frob}^2 ,$$

where  $X \in T_F$ ,  $g_{whip}$  is an arbitrary non-negative quadratic function of  $d(d-1)/2$  arguments, and  $C$  is an arbitrary symmetric non-negative definite  $d \times d$  matrix, and as usual  $P = FF^T$ . In particular, whip spin and plane motion are orthogonal under all left-invariant Riemannian metrics.

### 12.3 Left- and right-invariant metrics for frames

For any  $V \in O(d)$ , if  $F(t)$  is a path of frames, so is  $F(t)V$ , hence:

**Fact:**  $V \in O(d)$ ,  $X \in T_F \implies XV \in T_{FV}$ .

**Definition:** A Riemannian metric is right-invariant w.r.t.  $O(d)$  if it satisfies

$$g_{FV}(XV, YV) = g_F(X, Y) \quad \text{or, equivalently,} \quad g_{FV}(XV) = g_F(X),$$

for all  $V \in O(d)$ .

Right invariance is not in itself a far reaching property. It is, however, when combined with left-invariance. Therefore we make henceforth the

**Assumption:** The metric  $g_F(X, Y)$  (equivalently:  $g_F(X)$ ) is both left-invariant w.r.t.  $O(p)$  and right-invariant w.r.t.  $O(d)$ .

**Fact:** At  $F = E_d$ , we have  $g\left(\begin{pmatrix} V^T X_1 V \\ X_2 V \end{pmatrix}\right) = g\left(\begin{pmatrix} X_1 \\ X_2 \end{pmatrix}\right)$  for all  $V \in O(d)$ .

**Proof:** Let  $U = \begin{pmatrix} V & 0 \\ 0 & I_{p-d} \end{pmatrix}$  and note  $E_d V = U E_d$ . Hence:

$$g_{E_d}(X) = g_{E_d V}(XV) = g_{U E_d}(XV) = g_{E_d}(U^T X V) = g\left(\begin{pmatrix} V^T X_1 V \\ X_2 V \end{pmatrix}\right). \quad \square$$

The following theorem describes the consequences of left- and right-invariance for whip spin and plane motion separately:

**Theorem:**  $g\left(\begin{pmatrix} X_1 \\ 0 \end{pmatrix}\right) = \alpha_w \cdot \text{trace}(X_1^T X_1)$ ,  $g\left(\begin{pmatrix} 0 \\ X_2 \end{pmatrix}\right) = \alpha_p \cdot \text{trace}(X_2^T X_2)$ .

**Proof:** a) *Whip spin:* The whip spin matrix  $X_1$  is skew symmetric, hence there exists  $V \in O(d)$  that brings  $X_1$  to canonical form:

$$V^T X_1 V = \begin{pmatrix} 0 & -\sigma_1 & 0 & 0 & \dots \\ \sigma_1 & 0 & 0 & 0 & \dots \\ 0 & 0 & 0 & -\sigma_2 & \dots \\ 0 & 0 & \sigma_2 & 0 & \dots \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix},$$

where  $\sigma_i \geq 0$  are the canonical speeds in the invariant 2-D planes (they are also the singular values of  $X_1$ ). The  $2 \times 2$  blocks are the infinitesimal rotations that

generate whip spin. It is sufficient to consider  $X_1$ 's in canonical form. Denote with  $S_{ij}$  the infinitesimal rotation with unit speed in the  $i$ - $j$ -coordinate plane, that is,  $S_{ij}$  has action  $\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$  in this plane. We show that, for example,  $S_{12}$  and  $S_{34}$  are orthogonal under any left-right-invariant metric:  $g(S_{12}, S_{34}) = 0$  (where, by abuse of notation, we ignored the part due to plane motion). To this end, let  $V \in O(d)$  be a reflection that maps the 4'th coordinate to its negative, and note that  $V^T S_{12} V = S_{12}$  and  $V^T S_{34} V = -S_{34}$ . It follows

$$g(S_{12}, S_{34}) = g(V^T S_{12} V, V^T S_{34} V) = g(S_{12}, -S_{34}) = -g(S_{12}, S_{34}),$$

hence  $g(S_{12}, S_{34}) = 0$ . Therefore, the metric is of the form  $g(X_1) = \alpha_1 \sigma_1^2 + \alpha_2 \sigma_2^2 + \dots$ . The coefficients  $\alpha_i$  must be identical because there exists  $V$  such that, for example,  $V^T S_{12} V = S_{34}$ , implying permutation invariance in the  $\sigma_i$ 's. Thus,

$$g(X_1) = 2\alpha_w \cdot \sum_i \sigma_i^2 = \alpha_w \cdot \text{trace}(X_1^T X_1). \quad \square$$

b) *Plane motion*: Left-invariance implies  $g(X) = \text{trace}(X_2^T X_2 C)$  for some symmetric  $C$  of size  $(p-d) \times (p-d)$ . We show that right-invariance implies  $C = c \cdot I_d$ : From the above fact and a simple manipulation of the trace we get

$$\text{trace}(X_2^T X_2 C) = \text{trace}(V^T X_2^T X_2 V C) = \text{trace}(X_2^T X_2 V C V^T).$$

Hence  $V C V^T = C$  for all  $V \in SO(d)$ . By part c) of the corollary of Section 12.7 it follows  $C = c \cdot I_d$ .  $\square$

Here is a summary, written for arbitrary frames  $F$ , figuring in that whip spin and plane motion are orthogonal for left-invariant metrics:

**Corollary:** For some  $\alpha_w, \alpha_p > 0$ , and for  $P = F F^T$ , we have

$$\begin{aligned} g_F(F') &= \alpha_w \cdot \text{trace}(F'^T P F') + \alpha_p \cdot \text{trace}(F'^T (I - P) F') \\ &= \alpha_w \cdot \|F'^T F'\|_{Frob}^2 + \alpha_p/2 \cdot \|P'\|_{Frob}^2. \end{aligned}$$

**Proof:** The first equality is a straight translation from the frame  $E_d$  to an arbitrary frame  $F$ . The second equality requires a minor calculation at  $F = E_d$  and  $F'^T = (X_1^T, X_2^T)$ :

$$P' = F' F'^T + F F'^T = \begin{pmatrix} X_1 & 0 \\ X_2 & 0 \end{pmatrix} + \begin{pmatrix} X_1^T & X_2^T \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & X_2^T \\ X_2 & 0 \end{pmatrix},$$

because  $X_1$  is skew. Hence  $\|P'\|_{Frob}^2 = 2 \cdot \|X_2\|_{Frob}^2$ .  $\square$

## 12.4 Invariant metrics for planes

From what we have proven so far, it is almost trivial that invariant metrics on planes must be the plane speed component of the corollary in the previous section. We should note in what follows that metrics on planes are semi-metrics on frames because they are blind to motion within the plane, by definition. Because frames are the necessary ingredient for visual rendering, though, we prefer to think of plane metrics as semi-metrics on frames, for which we reserve the term ‘‘Grassmannian’’:

**Definition:** *A left- and right-invariant semi-metric  $g$  on frames is called a Grassmann semi-metric if it is invariant under dynamic within-plane rotations. That is, for any fixed frame such as  $E_d$ , any path  $V = V(t) \in SO(d)$  of whip rotations, and for all  $t$ ,*

$$g_{E_d V(t)}(E_d V'(t)) = 0 .$$

Because  $V'$  can be any skew matrix of size  $d \times d$ , we must have  $g_{whip} = 0$ , hence:

**Corollary:** *There exists essentially only one invariant Grassmann semi-metric:*

$$g_F(F') \propto \|P'\|_{Frob}^2 .$$

## 12.5 The equations of geodesics with regard to invariant metrics

We derive the equations for geodesic paths with regard to the invariant metrics of Section 12.4. In principle, any textbook of differential geometry, such as Spivak (1979, p. 444 and 451), has the equations for a geodesic in coordinates  $x_i$ :

$$\sum_l g_{kl} x_l'' = \frac{1}{2} \sum_{ij} \left( \frac{\partial}{\partial x_k} g_{ij} - \frac{\partial}{\partial x_i} g_{jk} - \frac{\partial}{\partial x_j} g_{ik} \right) x_i' x_j'$$

This form, however, is extremely unhelpful in our situation: Orthonormal  $d$ -frames in  $p$ -space form a submanifold of  $\mathbb{R}^{pd}$  defined in terms of implicit equations ( $F^T F = I_d$ ), endowed with Riemannian metrics that are not inherited from the Euclidean metric of  $\mathbb{R}^{pd}$  (except when  $\alpha_p = \alpha_p$ ). We re-cast the equations in a form that facilitates application to frames. The new equations will be in vectorized coordinates in order to avoid the tedium of index calculations. To this end, we introduce notation for directional derivatives in coordinates:

$$\partial_Y f(\mathbf{x}) = \sum_i Y_i \frac{\partial}{\partial x_i} f(x_1, x_2, \dots) ,$$

where  $\mathbf{x} = (x_1, x_2, \dots)^T$  are point coordinates and  $Y = (Y_1, Y_2, \dots)$  are coordinates of a tangent vector at  $\mathbf{x}$ . For a curve  $\mathbf{x}(t)$  the vector  $\mathbf{x}'$  also contains tangent vector coordinates. The reason for introducing directional derivatives is their ease of application



to vector-algebraic expressions. — For a Riemannian metric

$$g_{\mathbf{x}}(X, Z) = \sum_{ij} g_{ij}(\mathbf{x}) X_i Z_j$$

we can now make sense of  $\partial_Y g_{\mathbf{x}}$ :

$$(\partial_Y g_{\mathbf{x}})(X, Z) = \sum_{ij} (\partial_Y g_{ij}(\mathbf{x})) X_i Z_j .$$

With the aid of arbitrary tangent vectors  $Y$ , the equations for a geodesic can now be re-cast, first in indexed coordinates:

$$\sum_{k,l} g_{kl} x'_l Y_k = \frac{1}{2} \sum_{ijk} \left( \frac{\partial}{\partial x_k} g_{ij} - \frac{\partial}{\partial x_i} g_{jk} - \frac{\partial}{\partial x_j} g_{ik} \right) x'_i x'_j Y_k ,$$

then in vectorized coordinates:

$$\begin{aligned} g(Y, \mathbf{x}'') &= \frac{1}{2} [(\partial_Y g)(\mathbf{x}', \mathbf{x}') - (\partial_{\mathbf{x}'} g)(Y, \mathbf{x}') - (\partial_{\mathbf{x}'} g)(\mathbf{x}', Y)] \\ &= \frac{1}{2} (\partial_Y g)(\mathbf{x}', \mathbf{x}') - (\partial_{\mathbf{x}'} g)(Y, \mathbf{x}') , \end{aligned}$$

where we omitted the argument  $\mathbf{x}$  from  $g = g_{\mathbf{x}}$ . These equations are to hold for all  $Y$  denoting tangent vectors at  $\mathbf{x}$ .

From a purist's point of view of differential geometry, this form of the geodesic equations is peculiar: It looks like an invariant formulation but it isn't; for one thing,  $\mathbf{x}''$  does not denote a tangent vector, yet  $g(\mathbf{x}'', Y)$  is a well-defined algebraic expression. This form of the equations is just a device for executing messy coordinate calculations in vectorized notation. (In an invariant interpretation, the equations express the invariant geodesic condition  $D_X X = 0$  in the equivalent form  $g_{\mathbf{x}}(Y, D_X X) = 0$  for all tangent vectors  $Y$ , where  $X$  is the tangent vector denoted by  $\mathbf{x}'$ .)

We now derive the equations for geodesic paths of frames  $F = F(t)$  with regard to any of the invariant metrics. For the tangent vectors  $F'$  and  $Y$  at  $F$ , the metric in its bilinear form is

$$g_F(Y, F') = \alpha_p \cdot \text{trace}(Y^T F') + (\alpha_w - \alpha_p) \cdot \text{trace}(Y^T F F^T F') .$$

We calculate  $g(Y, F'')$ ,  $(\partial_Y g)(F', F')$  and  $(\partial_{F'} g)(Y, F')$  in turn:

$$\begin{aligned} g(Y, F'') &= \alpha_p \cdot \text{trace}(Y^T F'') + (\alpha_w - \alpha_p) \cdot \text{trace}(Y^T P F'') \\ (\partial_Y g)(F', F') &= (\alpha_w - \alpha_p) \cdot [\text{trace}(F'^T Y F^T F') + \text{trace}(F'^T F Y^T F')] \\ &= (\alpha_w - \alpha_p) \cdot 2 \cdot \text{trace}(Y^T F' F'^T F) \\ &= -(\alpha_w - \alpha_p) \cdot 2 \cdot \text{trace}(Y^T F' F^T F') \\ (\partial_{F'} g)(Y, F') &= (\alpha_w - \alpha_p) \cdot [\text{trace}(Y^T F' F^T F') + \text{trace}(Y^T F F'^T F')] \\ &= (\alpha_w - \alpha_p) \cdot \text{trace}(Y^T [F' F^T F' + F F'^T F']) \end{aligned}$$

We used repeatedly the trace identities

$$\begin{aligned}\partial_Y \text{trace}(AFF^T B) &= \text{trace}(AYF^T B) + \text{trace}(AFY^T B) , \\ \text{trace}(A^T) &= \text{trace}(A) , \quad \text{trace}(AB) = \text{trace}(BA) .\end{aligned}$$

For the second term we also made use of the fact that  $F^T F'$  is skew. The equations for geodesic paths are therefore:

$$\begin{aligned}0 &= g(Y, F'') - 1/2 \cdot (\partial_Y g)(F', F') + (\partial_{F'} g)(Y, F') \\ &= \text{trace} \left( Y^T \left[ \alpha_p \cdot F'' + (\alpha_w - \alpha_p) \cdot \{PF'' + 2F'F^T F' + FF'^T F'\} \right] \right)\end{aligned}$$

This is to hold for all tangent vectors  $Y$  at  $F$ . In order to flesh out this condition we need a lemma:

**Lemma:** For a  $p \times d$ -matrix  $Z$  the following conditions are equivalent:

- 1)  $\text{trace}(Y^T Z) = 0$  for all  $Y$  for which  $F^T Y$  is skew.
- 2)  $(I - P)Z = 0$  and  $F^T Z$  is symmetric.

**Proof** of the lemma: We can assume w.l.o.g.  $F = E_d$ , in which case

$$Z = \begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} , \quad Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} , \quad Y_1 = E_d^T Y \text{ is skew.}$$

The expression  $\text{trace}(Y^T Z)$  is just the Euclidean inner product on  $\mathbb{R}^{pd}$ . It is zero for all  $Y$  with skew  $Y_1$  iff  $Z_1$  is symmetric and  $Z_2 = 0$ . At an arbitrary frame  $F$ , this translates to  $F^T Z$  symmetric and  $(I - P)Z = 0$ .  $\square$

We apply the lemma to the equation for geodesic paths of frames: For

$$Z = \alpha_p \cdot F'' + (\alpha_w - \alpha_p) \cdot \{PF'' + 2F'F^T F' + FF'^T F'\}$$

we must have  $F^T Z$  symmetric and  $(I - P)Z = 0$ :

- The symmetry condition simplifies as follows: 1)  $F^T F' F^T F'$  is the square of the skew matrix  $F^T F'$ , which is symmetric; 2)  $F^T F F'^T F' = F'^T F'$ , which is also symmetric; 3)  $F^T P F'' = F^T F''$ . Hence  $F^T Z$  is symmetric iff  $F^T F''$  is symmetric.
- For the condition  $(I - P)Z = 0$ , note that  $(I - P)P F''$  and  $(I - P)F F'^T F'$  both vanish, hence

$$(I - P)Z = (I - P)(\alpha_p F'' + (\alpha_w - \alpha_p) 2 F' F^T F')$$

This proves the following:

**Theorem:** A path of frames  $F(t)$  is geodesic iff  $F^T F''$  is symmetric, and

$$(I - P)(\alpha_p F'' + 2(\alpha_w - \alpha_p) F' F^T F') = 0 ,$$

where  $\alpha_p$  and  $\alpha_w$  define a left- and right-invariant metric.

## 12.6 Construction of geodesics with regard to invariant metrics

The equations for geodesic paths of frames can be used to verify that a particular explicit construction yields geodesics. A dimensional argument will then confirm that this construction yields all geodesics.

Without loss of generality, we consider paths that start at the unit frame  $E_d$ , which we denote simply by  $E$  due to frequent use in this section. The functional form of paths we consider is

$$(*) \quad F(t) = \exp(St) E \exp(Qt), \quad S \text{ skew } p \times p, \quad Q \text{ skew } d \times d.$$

Hence  $\exp(St) \in SO(p)$  and  $\exp(Qt) \in SO(d)$  are both paths of rotations. The former transports the starting frame  $E$  through space starting at  $F(0) = E$ , while the latter modifies the whip spin of the path. It will be useful to have notation for the natural blocks of  $S$ :

$$S = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix},$$

where  $S_{11}$  is  $d \times d$ ,  $S_{12} = -S_{21}^T$  is  $d \times (p-d)$ , and  $S_{22}$  is  $(p-d) \times (p-d)$ . Note that  $S_{11} = E^T S E$ .

The goal is to apply the equations for geodesics to the paths  $F(t)$ . To this end the derivatives are:

$$\begin{aligned} F' &= \exp(St) (SE + EQ) \exp(Qt), \\ F'' &= \exp(St) (S^2 E + 2SEQ + EQ^2) \exp(Qt). \end{aligned}$$

As a first application we obtain the whip spin matrix:

$$F^T F' = \exp(-Qt) (S_{11} + Q) \exp(Qt)$$

Thus, the matrix  $Q$  can be used to modify the whip spin generated by the transport with  $\exp(St)$ .

**Proposition:** *Paths of frames of the form (\*) have constant speed:*

$$g_{F(t)}(F'(t)) = \alpha_w \|S_{11} + Q\|_{Frob}^2 + \alpha_p \|S_{21}\|_{Frob}^2$$

where  $\alpha_w$  and  $\alpha_p$  define a left- and right-invariant metric.

**Proof:** Calculate the contributions due to whip spin and plane motion and note that all terms  $\exp(St)$  and  $\exp(Qt)$  cancel:

$$\begin{aligned} \text{trace}((F^T F')^T (F^T F')) &= \text{trace}((S_{11} + Q)^T (S_{11} + Q)) = \|S_{11} + Q\|_{Frob}^2 \\ \text{trace}(F'^T (I - P) F') &= \text{trace}((SE + EQ)^T (I - EE^T) (SE + EQ)) \\ &= \text{trace}(E^T S^T (I - EE^T) S E) \\ &= \text{trace}((S_{11}^T, S_{21}^T) \begin{pmatrix} 0 & 0 \\ 0 & I_{p-d} \end{pmatrix} \begin{pmatrix} S_{11} \\ S_{21} \end{pmatrix}) \\ &= \text{trace}(S_{21}^T S_{21}) = \|S_{21}\|_{Frob}^2 \quad \square \end{aligned}$$

We now investigate the conditions under which the paths  $F(t)$  are geodesic for a given left- and right-invariant metric. The first condition is symmetry of  $F^T F''$ :

$$\begin{aligned} F^T F'' &= \exp(Qt)^T (E^T S^2 E + 2E^T S E Q + E^T E Q^2) \exp(Qt) \\ &= \exp(Qt)^T (-E^T S^T S E + 2S_{11} Q - Q^T Q) \exp(Qt) \end{aligned}$$

It follows that  $F^T F''$  is symmetric iff  $S_{11} Q$  is symmetric:

$$S_{11} Q = (S_{11} Q)^T = Q^T S_{11}^T = Q S_{11} ,$$

That is, we have symmetry iff  $S_{11}$  and  $Q$  commute.

We turn to the second condition for geodesic paths which requires calculation of  $(I - P)F''$  and  $(I - P)F' F^T F'$ :

$$\begin{aligned} (I - P)F'' &= \exp(St) (I - E E^T) (S^2 E + 2S E Q + E Q^2) \exp(Qt) \\ &= \exp(St) (I - E E^T) S (S E + 2E Q) \exp(Qt) \\ &= \exp(St) \begin{pmatrix} 0 & 0 \\ 0 & I_{p-d} \end{pmatrix} \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix} \begin{pmatrix} S_{11} + 2Q \\ S_{21} \end{pmatrix} \exp(Qt) \\ &= \exp(St) \begin{pmatrix} 0 \\ S_{21}(S_{11} + 2Q) + S_{22} S_{21} \end{pmatrix} \exp(Qt) \\ (I - P)F' F^T F' &= \exp(St) (I - E E^T) (S E + E Q) E^T (S E + E Q) \exp(Qt) \\ &= \exp(St) (I - E E^T) S E E^T (S E + E Q) \exp(Qt) \\ &= \exp(St) (I - E E^T) S E (S_{11} + Q) \exp(Qt) \\ &= \exp(St) \begin{pmatrix} 0 & 0 \\ 0 & I_{p-d} \end{pmatrix} \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix} \begin{pmatrix} S_{11} + Q \\ 0 \end{pmatrix} \exp(Qt) \\ &= \exp(St) \begin{pmatrix} 0 \\ S_{21}(S_{11} + Q) \end{pmatrix} \exp(Qt) \end{aligned}$$

Combining the two terms, we get the second condition for geodesic paths: We have  $(I - P)(\alpha_p F'' + 2(\alpha_w - \alpha_p) F' F^T F') = 0$  iff

$$\alpha_p [S_{21}(S_{11} + 2Q) + S_{22} S_{21}] + 2(\alpha_w - \alpha_p) S_{21}(S_{11} + Q) = 0 .$$

Rearranging this equation yields:

**Theorem:** *A path of frames of the form (\*) is geodesic iff  $S_{11}$  and  $Q$  commute and*

$$\alpha_p S_{22} S_{21} + S_{21} ((2\alpha_w - \alpha_p) S_{11} + 2\alpha_w Q) = 0 ,$$

where  $\alpha_p$  and  $\alpha_w$  define a left- and right-invariant metric.

**Corollary:** *The following are sufficient conditions for paths of frames of the form (\*) to be geodesic:*

- $S_{22} = 0$  and  $Q = (\frac{\alpha_p}{2\alpha_w} - 1) S_{11}$ , hence:

$$F(t) = \exp\left(\begin{pmatrix} S_{11} & S_{12} \\ S_{21} & 0 \end{pmatrix} t\right) \begin{pmatrix} \exp((\frac{\alpha_p}{2\alpha_w} - 1) S_{11}t) \\ 0 \end{pmatrix}$$

These paths generate all possible geodesics emanating from  $E$  for the left- and right-invariant metric defined by  $\alpha_p$  and  $\alpha_w$ .

- $S_{21} = 0$ , and  $S_{11}$  and  $Q$  commute; that is, these paths are pure whip spin:  $F = E \exp((S_{11} + Q)t)$ ; they are universally geodesic for all left- and right-invariant metrics.
- $S_{11} = Q = 0$  and  $S_{22} = 0$ , that is, these paths are pure plane motion; they are universally geodesic for all left- and right-invariant metrics.

The conditions follow immediately from the theorem. The family of geodesics following from the first criterion is complete because the dimension of skew matrices  $S$  that satisfy the criterion equals the dimension of the Stiefel manifold  $V_{d,p}$ , namely,  $pd - (d+1)d/2$ .

The second and third criteria generate universally geodesic paths because the criteria are independent of  $\alpha_w$  and  $\alpha_p$ .  $\square$

Note a couple of special cases:

- For  $\alpha_p = 2$ ,  $\alpha_w = 1$  the first criterion yields  $Q = 0$ , hence the geodesic paths are of the form  $F(t) = \exp(St)E$ . The speed measure is  $g_F(F') = \|S\|_{Frob}^2$  due to  $S_{22} = 0$ .
- For  $\alpha_p = \alpha_w = 1$  the first criterion yields  $Q = -S_{11}/2$ , hence the geodesic paths are of the form  $F(t) = \exp(St)E \exp(-S_{11}t/2)$ . The speed measure is  $g_F(F') = \|SE\|_{Frob}^2$  due to  $S_{22} = 0$ .

## 12.7 $O(p)$ -Invariant real matrices

We prove a theorem from linear algebra that underlies the invariance theorems for Riemannian metrics on Stiefel manifolds.

We consider a  $p$ -dimensional *real* vector space with inner product and linear maps from the vector space into itself. In particular, let  $\mathcal{U}$  be a set of linear maps and  $A$  an individual linear map; assume that  $A$  commutes with all elements of  $\mathcal{U}$ :  $AU = UA$  for all  $U \in \mathcal{U}$ . The question is under what conditions we can infer that  $A$  is a multiple of the identity:  $A = c \cdot I_p$ .

**Theorem:** 1) If for any two unit vectors  $\mathbf{x}$  and  $\mathbf{y}$  there exists  $U \in \mathcal{U}$  such that  $U\mathbf{x} = \mathbf{y}$ , and if  $A$  has a real eigenvalue, then  $A = c \cdot I_p$ .

2) If for any two unit vectors  $\mathbf{x}$  and  $\mathbf{y}$  there exists  $U \in \mathcal{U}$  such that  $U\mathbf{x} = \mathbf{y}$  and  $U\mathbf{y} = \mathbf{x}$ , then  $A = c \cdot I_p$ .

**Proof:** 1) The map  $A$  has a real eigenvalue  $\lambda$  by assumption. Let  $\mathbf{x}$  be a unit length eigenvector for  $\lambda$  and  $\mathbf{y}$  an arbitrary unit vector. Again by assumption  $\mathbf{y} = U\mathbf{x}$  for some  $U \in \mathcal{U}$ , so  $A\mathbf{y} = AU\mathbf{x} = UA\mathbf{x} = \lambda U\mathbf{x} = \lambda\mathbf{y}$ .

2) A general linear map  $A$  has a complex eigenvalue  $\lambda = \lambda_r + i\lambda_i$  over the complexification of the linear space and there exists a complex eigenvector  $\mathbf{x} = \mathbf{x}_r + i\mathbf{x}_i$ :  $A\mathbf{x} = \lambda\mathbf{x}$ . Because  $A$  is real, complex conjugation yields another eigenvalue and eigenvector:  $A\bar{\mathbf{x}} = \bar{\lambda}\bar{\mathbf{x}}$ . It follows that the real vectors  $\mathbf{x}_r$  and  $\mathbf{x}_i$  span an invariant real plane in which  $A$  acts as a rotation and dilation:

$$A(\mathbf{x}_r, \mathbf{x}_i) = (\mathbf{x}_r, \mathbf{x}_i) \begin{pmatrix} c_\phi & s_\phi \\ -s_\phi & c_\phi \end{pmatrix} r$$

where  $\lambda = (c_\phi + is_\phi)r$  and  $\mathbf{x}_r$  and  $\mathbf{x}_i$  are unit length. See Halmos 1958, p. 164 for a derivation in a related case.

By assumption there exists  $U \in \mathcal{U}$  such that  $U\mathbf{x}_r = \mathbf{x}_i$  and  $U\mathbf{x}_i = \mathbf{x}_r$ . Because  $U$  commutes with  $A$  we have

$$AU(\mathbf{x}_r, \mathbf{x}_i) = A(\mathbf{x}_i, \mathbf{x}_r) = (\mathbf{x}_r, \mathbf{x}_i) \begin{pmatrix} s_\phi & c_\phi \\ c_\phi & -s_\phi \end{pmatrix} r,$$

$$UA(\mathbf{x}_r, \mathbf{x}_i) = (\mathbf{x}_i, \mathbf{x}_r) \begin{pmatrix} c_\phi & s_\phi \\ -s_\phi & c_\phi \end{pmatrix} r = (\mathbf{x}_r, \mathbf{x}_i) \begin{pmatrix} -s_\phi & c_\phi \\ c_\phi & s_\phi \end{pmatrix} r.$$

It follows  $s_\phi = 0$ , hence  $\phi = 0$  or  $\phi = \pi$ , and finally  $A = \pm rI_p$ . Thus we showed the existence of a real eigenvalue, and part 1) applies.  $\square$

**Corollary:** *The matrix  $A$  is a multiple of the identity if one of the following holds:*

- a)  $p \geq 2$  and  $A$  commutes with all elements of  $O(p)$ ;
- b)  $p \geq 3$  and  $A$  commutes with all elements of  $SO(p)$ .
- c)  $p \geq 2$  and  $A$  is symmetric and commutes with all elements of  $SO(p)$ .

**Proof:** a) The set  $\mathcal{U} = O(p)$  satisfies the assumption of part 2) of the Theorem because it contains reflections that can map any pair of unit vectors onto each other.

b) In order to apply part 2) of the Theorem to  $\mathcal{U} = SO(p)$  one needs  $p \geq 3$ : For two unit vectors  $\mathbf{x}$  and  $\mathbf{y}$  consider the reflection that maps  $\mathbf{x}$  and  $\mathbf{y}$  onto each other in the plane spanned by  $\mathbf{x}$  and  $\mathbf{y}$ ; the determinant of this  $2 \times 2$  reflection is -1. Pick an arbitrary vector  $\mathbf{z}$  orthogonal to  $\mathbf{x}$  and  $\mathbf{y}$  and extend the reflection to a map on the full  $p$ -dimensional space by requiring  $\mathbf{z} \leftarrow -\mathbf{z}$  and leaving the space orthogonal to  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  fixed; the resulting map is orthogonal and has determinant +1. Note that a third dimension (represented by  $\mathbf{z}$ ) was needed to correct the determinant to +1.

c) Symmetric  $A$ 's have real eigenvalues, so part 1) of the Theorem applies.  $\square$

**Fact:** *For  $p = 2$ , if  $A$  commutes with all elements of  $SO(2)$ , then  $A \in \mathbb{R} \cdot SO(2)$ .*

**Proof:** By part 1) of the Theorem, if  $A$  has a real eigenvalue then  $A = c \cdot I_2$ . If  $A$  has a complex eigenvalue, then the first half of the proof of part 2) of the Theorem applies:  $A$  is a multiple of a rotation.  $\square$

This last fact explains the exceptions that arise in two dimensions if one insists on  $SO(2)$ -invariance as opposed to  $O(2)$ -invariance.

## 12.8 Removing whip spin

We prove theorem 1 of Section 7.3 by constructing for a given path of frames  $\tilde{F}(t)$  another path  $F(t)$  that has zero whip spin and generates the same path of planes:  $\tilde{F}(t) = F(t)V(t)$  for some path  $V(t) \in SO(d)$ .

The proof is by deriving a differential equation for  $V(t)$  from the condition of vanishing whip spin for  $F(t) = \tilde{F}(t)V(t)^T$ :

$$0 = F^T F' = V \tilde{F}^T (\tilde{F} V^T)' = V \tilde{F}^T (\tilde{F}' V^T + \tilde{F} V'^T) = V S V^T + V V'^T,$$

where  $S(t) = \tilde{F}^T \tilde{F}'$  is the whip spin matrix of  $\tilde{F}$ . Dropping the factor  $V$ , transposing, and using skewness of  $S$ , we get the matrix differential equation

$$V S = V',$$

which can be interpreted as a vector differential equation for the rows of  $V$ . By basic existence theorems we obtain a matrix path  $V(t)$  satisfying the initial conditions  $V(0) = I_d$ . It remains to show that  $V(t) \in SO(d)$ :

$$(V V^T)' = V' V^T + V V'^T = V S V^T + V S^T V^T = 0$$

due to skewness of  $S$ . Therefore  $V V^T$  is constant and equal to  $V(0)V(0)^T = I_d$ .  $\square$

## 12.9 Data rotation versus frame motion

We prove the fact mentioned at the end of Section 9.4:

$$\|U'\|_{Frob}^2 = \|P F'\|_{Frob}^2 + 2 \cdot \|(I - P)F'\|_{Frob}^2,$$

where  $U(t) = (F(t), G(t))$  is a path in  $SO(p)$ , decomposed into a  $d \times p$ -frame  $F(t)$  and a  $(p - d) \times p$ -frame  $G(t)$  with zero-whip spin. The following trivial calculations decompose the (essentially unique) speed measure  $\|U'\|_{Frob}^2$  of data rotations:

$$\begin{aligned} \|U'\|_{Frob}^2 &= \|U^T U'\|^2 = \left\| \begin{pmatrix} F^T F' & F^T G' \\ G^T F' & G^T G' \end{pmatrix} \right\|_{Frob}^2 \\ &= \|F^T F'\|_{Frob}^2 + \|F^T G'\|_{Frob}^2 + \|G^T F'\|_{Frob}^2 + \|G^T G'\|_{Frob}^2 \\ &= \|F^T F'\|_{Frob}^2 + 2 \cdot \|G^T F'\|_{Frob}^2 + \|G^T G'\|_{Frob}^2 \end{aligned}$$

The last equation follows because  $U^T U' + U'^T U = 0$ , hence  $F^T G' = -(G^T F')^T$ . We note that  $G G^T = I - P$ , where as always  $P = F F^T$ . Therefore

$$\|G^T F'\|_{Frob}^2 = \text{trace}(F'^T G G^T F') = \text{trace}(F'^T (I - P) F') = \|(I - P) F'\|_{Frob}^2,$$

which is just the term for plane speed in theorem 2 of Section 9.3. The whip spin matrix  $G^T G'$  is zero by assumption.  $\square$



## References

- [1] Andrews, D. F. (1972), “Plots of High-Dimensional Data,” *Biometrics* **28**, pp. 125–136.
- [2] Asimov, D. (1985), “The grand tour: a tool for viewing multidimensional data,” *SIAM J. Sci. Statist. Computing* **6** 1, pp. 128–143.
- [3] Asimov, D., and Buja, A. (1994), “The grand tour via geodesic interpolation of 2-frames,” in *Visual Data Exploration and Analysis, Symposium on Electronic Imaging Science and Technology*, IS&T/SPIE (Soc. for Imaging Sci. and Technology/Internat. Soc. for Optical Engineering).
- [4] Bjorck, A., and Golub, G. H. (1973), “Numerical methods for computing angles between linear subspaces,” *Mathematics of Computation* **27** 123, pp. 579–594.
- [5] Buja, A., and Asimov, D. (1986), “Grand tour methods: an outline,” *Computer Science and Statistics: Proc. of the 17th Symp. on the Interface between Comput. Sci. and Statist.*, Amsterdam: Elsevier, pp.63–67.
- [6] Buja, A., Asimov, D., Hurley, C., and McDonald, J. A. (1988), “Elements of a viewing pipeline for data analysis,” in *Dynamic Graphics for Statistics*, eds. W. S. Cleveland and M. E. McGill, Belmont, CA: Wadsworth, pp. 277-308.
- [7] Buja, A., Hurley, C., and McDonald, J. A. (1986), “A data viewer for multivariate data,” *Computer Science and Statistics: Proc. of the 18th Symp. on the Interface between Comput. Sci. and Statist.*, Amsterdam: Elsevier.
- [8] Buja, A., Cook, D., and Swayne, D. F. (1996), “Interactive high-dimensional data visualization,” *Journal of Computational and Graphical Statistics* **5**, pp. 78–99.
- [9] Buja, A., Cook, D., Asimov, D., and Hurley, C. (2004), “Computational Methods for High-Dimensional Rotations in Data Visualization,” to appear in *Handbook of Statistics: Data Mining and Computational Statistics*, eds. C. R. Rao and E. Wegman; Amsterdam: North-Holland.
- [10] Carr, D. B., Littlefield, R. J., and Nicholson, W. L. (1986), “Scatterplot matrix techniques for large N,” *Computer Science and Statistics: Proc. of the 17th Symp. on the Interface between Comput. Sci. and Statist.*, Amsterdam: Elsevier, pp. 297–306.
- [11] Carr, D. B. (1991), “Looking at large data sets using binned data plots,” in: *Computing and Graphics in Statistics*, eds. A. Buja and P. A. Tukey, pp. 7–39, New York: Springer.

- [12] Carr, D. B., Wegman, E. J., Luo, Q. (1996), “ExploreN: Design considerations past and present,” Technical Report 129, Center for Computational Statistics, George Mason University, Fairfax, VA 22030.
- [13] Chambers, J. M., Cleveland, W. S., Kleiner, B., and Tukey, P. A. (1983), *Graphical Methods for Data Analysis*, Belmont, CA: Wadsworth.
- [14] Conway, J. H., Hardin, R. H., and Sloane, N. J. A. (1996), “Packing lines, planes, etc.: Packings in Grassmannian spaces,” *Journal of Experimental Mathematics* **5**, pp. 139–159.
- [15] Cook, D., and Buja, A. (1996), “Manual controls for high-dimensional data projections,” technical report, Iowa State University and AT&T Laboratories.
- [16] Cook, D., Buja, A., Cabrera, J., and Hurley, H. (1995), “Grand tour and projection pursuit,” *J. of Computational and Graphical Statistics* **2** 3, pp. 225–250.
- [17] Cook, D. R., and Weisberg, S. (1994), *An Introduction to Regression Graphics*, New York: Wiley.
- [18] Donoho, D. L., Huber, P. J., Ramos, E. and Thoma, M. (1982), “Kinematic Display of Multivariate Data,” in *Proc. of the Third Annual Conference and Exposition of the National Computer Graphics Association*.
- [19] Duffin, K. L., and Barrett, W. A. (1994), “Spiders: a new user interface for rotation and visualization of N-dimensional point sets,” in *Proceedings Visualization '94*, IEEE Computer Society Press, Los Alamitos, California, pp. 205–211.
- [20] Friedman, J. H. (1987), “Exploratory projection pursuit,” *Journal of the American Statistical Association*, **82**, pp. 249–266.
- [21] Furnas G. W., and Buja A. (1994), “Prosection Views: Dimensional Inference through Sections and Projections,” *Journal of Computational and Graphical Statistics*, **3**, pp. 323–385.
- [22] Golub, G. H., and Van Loan, C. F. (1983), *Matrix Computations*, second edition, Baltimore, Maryland: The Johns Hopkins University Press.
- [23] Halmos, P. R. (1958), *Finite-Dimensional Vector Spaces*, New York: Springer.
- [24] Halmos, P. R. (1970), “Finite-Dimensional Hilbert Spaces,” *The American Mathematical Monthly*, **77** 5, pp. 457–464.
- [25] Hanson, A. J., and Heng, P. A. (1991), “Visualizing a fourth dimension using geometry and light,” *Proceedings Visualization '91*, IEEE Computer Society Press, Los Alamitos, California, pp. 321–328.

- [26] Hurley, C. (1987), *The Data Viewer: A Program for Graphical Data Analysis*, PhD Thesis and Tech. Report, Statistics Dept., University of Washington, Seattle.
- [27] Hurley, C., and Buja, A. (1990), “Analyzing high-dimensional data with motion graphics,” *SIAM Journal on Scientific and Statistical Computing*, **11** 6, pp. 1193–1211.
- [28] Kobayashi, S., and Nomizu, K. (1969), *Foundations of Differential Geometry*, vol. II, New York: Springer.
- [29] Inselberg, A. (1985), “The plane with parallel coordinates,” *The Visual Computer* 1, New York: Springer, pp. 69–91.
- [30] Littman, M., Swayne, D. F., Dean, N., and Buja, A. (1992), “Visualizing the embedding of objects in Euclidean space,” *Computing Science and Statistics: Proc. of the 24th Symp. on the Interface*, Fairfax Station, VA: Interface Foundation of North America, Inc., pp. 208–217.
- [31] Miller, J. J., and Wegman, E. J. (1991), “Construction of line densities for parallel coordinate plots,” in: *Computing and Graphics in Statistics*, eds. A. Buja and P. A. Tukey, pp. 107–123, New York: Springer.
- [32] McDonald, J. A. (1982), “Orion I: Interactive graphics for data analysis,” in *Dynamic Graphics for Statistics*, eds. W. S. Cleveland and M. E. McGill, Belmont, CA: Wadsworth.
- [33] Scott, D. W. (1985), “Average shifted histograms: effective non-parametric density estimation in several dimensions,” *Ann. of Statist.*, **13**, pp. 1024–1040.
- [34] Scott, D. W. (1992), *Multivariate Density Estimation: Theory, Practice, and Visualization*, New York, NY: Wiley.
- [35] Scott, D. W. (1995), “Incorporating density estimation into other exploratory tools,” *ASA 1995 Proceedings of the Section on Statistical Graphics*, pp. 28–35.
- [36] Spivak, M. (1979), *Differential Geometry*, volume I, New York, NY: Wiley.
- [37] Swayne, D. F., Cook, D., and Buja, A. (1998), “XGobi: Interactive Dynamic Data Visualization in the X Window System,” *Journal of Computational and Graphical Statistics*, **7** 1, pp. 113–130.
- [38] Swayne, D.F., Buja, A., Temple-Lang, D. (2003), “Exploratory Visual Analysis of Graphs in GGobi,” Proceedings of the *Third Annual Workshop on Distributed Statistical Computing* (DSC 2003), Vienna.

- [39] Swayne, D.F., Temple-Lang, D., Buja, A., and Cook, D. (2002), “GGobi: Evolving from XGobi into an Extensible Framework for Interactive Data Visualization,” *Journal of Computational Statistics and Data Analysis*.
- [40] Tierney, L. (1990), *Lisp-Stat*, New York, NY: Wiley.
- [41] Tukey, J. W. (1987), “Comment on ‘Dynamic graphics for data analysis’ by Becker et al.,” *Statistical Science*, **2** 355-395; also in *Dynamic Graphics for Statistics*, eds. W. S. Cleveland and M. E. McGill, Belmont, CA: Wadsworth.
- [42] Tukey, J. and Tukey, P. (1981), “Graphical display of data sets in 3 or more dimensions,” in: *Interpreting Multivariate Data*, ed. V. Barnett, New York: Wiley.
- [43] Tukey, J. and Tukey, P. (1990), “Strips Displaying Empirical Distributions: I. Textured Dot Strips,” Bellcore Technical Memorandum.
- [44] Wegman, E. J. (1991), “The grand tour in k-dimensions,” *Computing Science and Statistics: Proceedings of the 22nd Symposium on the Interface*, 127–136.
- [45] Wegman, E. J. (2003), “Visual data mining,” *Statistics in Medicine*, **22**, 1383–1397, plus 10 color plates.
- [46] Wegman, E. J., and Carr, D. B. (1993), “Statistical graphics and visualization,” in *Handbook of Statistics 9: Computational Statistics*, 857–958, ed. C. R. Rao; Amsterdam: Elsevier.
- [47] Wegman, E. J. and Shen J. (1993), “Three-dimensional Andrews plots and the grand tour,” *Computing Science and Statistics*, **25**, 284–288.
- [48] Wegman, E. J., Poston, W. L., and Solka, J. L. (1998), “Image grand tour,” *Automatic Target Recognition VIII - Proceedings of SPIE*, 3371, 286–294.
- [49] Symanzik, J., Wegman, E., Braverman, A., and Luo, Q. (2002), “New applications of the image grand tour,” *Computing Science and Statistics*, **34**, 500–512.
- [50] Wegman, E. J., and Solka, J. L. (2002). “On some mathematics for visualizing high dimensional data,” *Sanhkya (A)*, **64** (2), 429–452.
- [51] Wegman, E. J., and Luo, Q. (1996), “High dimensional clustering using parallel coordinates and the grand tour,” Technical report No. 124, Center for Computational Statistics, George Mason University.
- [52] Wong, Y.-C. (1967), “Differential geometry of Grassmann manifolds,” *Proc. of the Nat. Acad. of Sci.*, **57**, 589-594.

- [53] Young, F. W., Kent, D. P., and Kuhfeld, W. F. (1988), “Dynamic graphics for exploring multivariate data,” in: *Dynamic Graphics for Statistics*, eds. W. S. Cleveland and M. E. McGill, Belmont, CA: Wadsworth.
- [54] Young, F. W., and Rheingans, P. (1991), “High-dimensional depth-cuing for guided tours of multivariate data,” in: *Computing and Graphics in Statistics*, eds. A. Buja and P. A. Tukey, pp. 239–252, New York: Springer.