

# On Interactive Visualization of High-dimensional Data using the Hyperbolic Plane

Jörg A. Walter · Helge Ritter

Neuroinformatics Group · Department of Computer Science  
University of Bielefeld · D-33615 Bielefeld · Germany

walter@techfak.uni-bielefeld.de

## ABSTRACT

We propose a novel projection based visualization method for high-dimensional datasets by combining concepts from MDS and the geometry of the hyperbolic spaces. Our approach *Hyperbolic Multi-Dimensional Scaling* (H-MDS) extends earlier work [7] using hyperbolic spaces for visualization of tree structures data (“hyperbolic tree browser”).

By borrowing concepts from multi-dimensional scaling we map proximity data directly into the 2-dimensional hyperbolic space (H2). This removes the restriction to “quasi-hierarchical”, graph-based data – limiting previous work. Since a suitable distance function can convert all kinds of data to proximity (or distance-based) data this type of data can be considered the most general.

We used the circular Poincaré model of the H2 which allows effective human-computer interaction: by moving the “focus” via mouse the user can navigate in the data without losing the “context”. In H2 the “fish-eye” behavior originates not simply by a non-linear view transformation but rather by extraordinary, non-Euclidean properties of the H2. Especially, the exponential growth of length and area of the underlying space makes the H2 a prime target for mapping hierarchical and (now also) high-dimensional data.

We present several high-dimensional mapping examples including synthetic and real world data and a successful application for unstructured text. By analyzing and integrating multiple film critiques from news:rec.art.movies.reviews and the internet movie database, each movie becomes placed within the H2. Here the idea is, that related films share more words in their reviews than unrelated. Their semantic proximity leads to a closer arrangement. The result is a kind of high-level content structured display allowing the user to explore the “space of movies”.

**Keywords:** visualizing high-dimensional data, infoViz, hyperbolic multi-dimensional scaling, H-MDS, focus+context, semantic browsing, text mining.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGKDD '02 Edmonton, Alberta, Canada

Copyright 2002 ACM 1-58113-567-X/02/0007 ...\$5.00.

## 1. INTRODUCTION

For many tasks of exploratory data analysis visualization plays an important role. It is a key for efficient integration of human expertise – not only to include his background knowledge, intuition and creativity, but also his powerful pattern recognition and processing capabilities. The design goals for an optimal user interaction strongly depend on the given visualization task but they certainly include an easy and intuitive navigation with strong support for the user's orientation.

Since most of available data display devices are two-dimensional – paper and screens – the following problem must be solved: finding a meaningful spatial mapping of data onto the display area. One limiting factor is the “restricted neighborhood” around a point in a Euclidean 2D surface. *Hyperbolic spaces* open an interesting loophole. The extraordinary property of exponential growth of neighborhood with increasing radius – around all points – enables one to build novel displays.

The “*hyperbolic tree viewer*”, developed at Xerox Parc [7], demonstrates the remarkable elegant interactive capabilities (visit <http://www.inxight.com> for a Java demo). The hyperbolic model appears as a continuously graded, focus+context mapping to the display. Very quickly the navigation by mouse clicks and drags appear natural and intuitive. It supports up to 10 times as many nodes as conventional approaches while providing more effective navigation around the tree-hierarchy.

The question how effective is visualization and navigation in the hyperbolic space was studied by Pirolli et al. [12]. By conducting eye-tracker experiments they found that the focus+context navigation can significantly accelerate the “information foraging”. Risken et al. [14] compared traditional and hyperbolic browsers and found significant improvement in task time for this novel display type.

Unfortunately, previous usage of *direct* hyperbolic visualization was limited to hierarchical, tree-like, or “quasi-graph” data. A H2 grid layout of data was recently introduced by one of the authors by generalizing Kohonen's Self-Organizing Map algorithm (HSOM) to the hyperbolic space [15]. Jörg Ontrup applied this HSOM successfully to text categorization and browsing [11].

In this work we go one step further and introduce the *hyperbolic multi-dimensional scaling* (H-MDS) for a direct construction of a distance preserving embedding of high-dimensional data into the hyperbolic space. The resulting H-MDS combines a number of attractive features of multi-

dimensional scaling and hyperbolic spaces for an interactive display. Compared to the HSOM it is not based on a discrete grid and can be used also in situations where only dissimilarity data but not a vector representation is available.

In Sec. 2 and 3 we introduce the hyperbolic space and the standard multi-dimensional scaling. Sec. 4 explains the combination of both concepts into H-MDS. Even though the look and feel of an interactive visualization and navigation is hardly compressible to paper format, we report several results and snapshots of first experiments in Sec. 5. In Sec. 6 we apply the H-MDS for spanning a space of unstructured text documents, i.e., special “averaged” film critiques. This approach allows to navigate in the space of selected movies.

## 2. THE HYPERBOLIC SPACE H2

2300 years ago, the Greek mathematician Euclid founded his geometry on five axioms. The fifth, the “parallel axiom”, appeared unnecessary to many of his colleagues. And they tried hard to prove it derivable – without success. After almost 2000 years Lobachevsky (1793-1856), Bolyai (1802-1860), and Gauss (1777-1855) negated the axiom and independently discovered the non-Euclidean geometry. There exist only two geometries with constant non-zero curvature. Through our sight of common spherical surfaces (e.g. earth, orange) we are familiar with the *spherical geometry* and its constant *positive* curvature. Its counterpart with constant *negative* curvature is known as the *hyperbolic plane* H2 (with analogous generalizations to higher dimensions) [2, 18]. Unfortunately, there is no “good” embedding of the H2 in  $\mathbb{R}^3$ , which makes it harder to grasp the unusual properties of the H2. Local patches resemble the situation at a saddle point, where the neighborhood grows faster than in flat space (see Fig. 1). Standard textbooks on Riemannian geometry (see, e.g. [8]) examine the relationship and expose that the circumference  $c$  and area  $a$  of a circle of radius  $r$  are given by

$$\text{area: } a(r) = 4\pi \sinh^2(r/2) \quad (1)$$

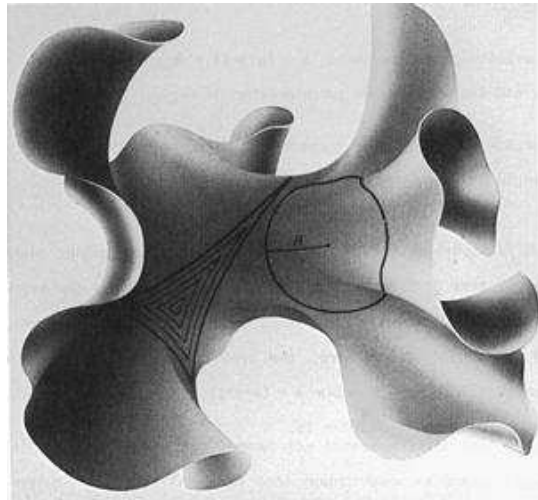
$$\text{circumference: } c(r) = 2\pi \sinh(r) . \quad (2)$$

This bears two remarkable asymptotic properties, (i) for small radius  $r$  the space “looks flat” since  $a(r) \approx \pi r^2$  and  $c(r) \approx 2\pi r$ . (ii) For larger  $r$  both grow *exponentially* with the radius. As observed in [7, 6], this trait makes the hyperbolic space ideal for embedding hierarchical structures. Fig. 1 illustrates the spatial relations by embedding a small patch of the H2 in  $\mathbb{R}^3$ .

To use the visualization potential of the H2 we must solve two problems: (i) the data must be “accommodated” in the hyperbolic space (see Sec. 4) and (ii) in order to inspect the result we need a projection of the H2 onto a suitable display. For practical and technological reasons this is a “flat surface” (we will not be able to buy a hyperbolic screen – at least for the foreseeable future). Fortunately, the projection problem was solved more than a century ago.

### 2.1 Projections of the Hyperbolic Space H2

It lays in the nature of a curved space to resist the *perfect* projection into the flat Euclidean surface. Each attempt compromises one or more correct representations of length, area, and angle (form) relations, as is well known from the spherical geometry (e.g., the Mercator, Lambert, and stereographic projection). Similarly to the spherical geometry



**Figure 1:** There is literally more room in hyperbolic space than in Euclidean space, as shown in this illustrated embedding of the hyperbolic plane into 3D Euclidean space (from [19], courtesy of Jeffrey Weeks). (Right:) Exponential growth (Eq. 1) of the circumference  $c(r)$  and area  $a(r)$  is experienced if a “circle” with radius  $r$  is drawn in the wrinkling structure. (Left:) The sum of angles in a triangle is smaller than  $180^\circ$ . A H2 paper model of the H2 can be made by gluing many equilateral triangles along their edges in such a way that seven triangles meet at each vertex.

several mappings were developed, four are especially well examined: (i) the *Minkowski*, (ii) the *upper-half plane*, (iii) the *Klein-Beltrami*, and (iv) the *Poincaré* or *disk* mapping. See [2] for more details and geometric mapping to convert in-between (i)–(iv).

### 2.2 Properties of the Poincaré Projection

What makes the Poincaré projection for our purpose the most suitable?

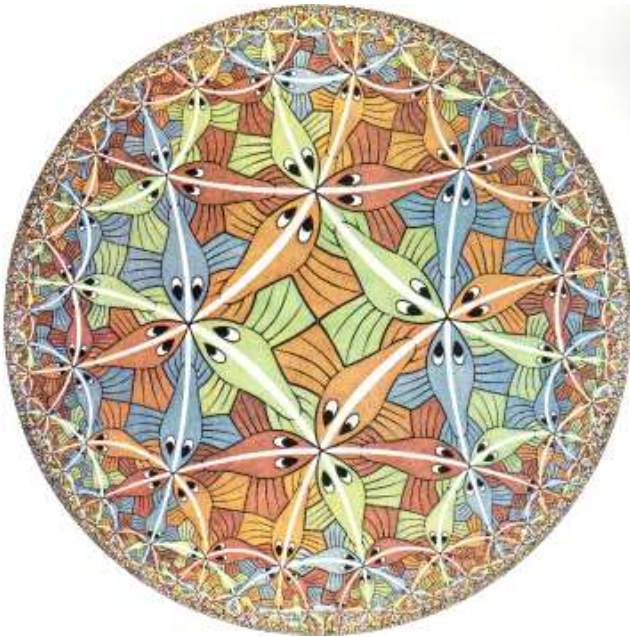
**Display compatibility:** The infinite large area of the H2 is mapped entirely into a circle, the Poincaré disk *PD*. This infinity representation fascinated M. Escher and inspired him to Fig. 2.

**Circle rim “=  $\infty$ ” :** All remote points are close to the rim, without touching it.

**Focus+Context:** The *focus* can be moved to each location in H2, like a “fovea”. The zooming factor is 0.5 in the center and falls (exponentially) off with distance to the fovea. Therefore, the context appears very natural. As more remote things are, the less spatial representation is assigned in the current display.

**Lines become circles:** All H2-lines<sup>1</sup> appear as *circle* arc segments or (centered) straight lines in *PD*. There extensions cross the *PD*-rim always perpendicular on

<sup>1</sup>A *line* is by definition the shortest path between two points



**Figure 2: Woodcut by Maurits Escher, named “Circle Limit III” (1958).** After seeing Coxeter’s picture [2] of the Poincaré projection of the H2, the artist was fascinated by the infinite space covered precisely in the disk. All the white arcs are lines in H2 which (should) intersect perpendicular at the rim (some are not perfectly drawn). Note the “fish-eye” effect, seen at the fishes: *all* are of equal size in H2 – but they *appear* larger in the center.

both ends<sup>2</sup>, see Fig. 2).

**Conformal mapping:** Angles (and therefore form) relations are preserved in  $PD$ , area and length relations obviously not (in contrast, e.g., to the Klein-Beltrami model which is length preserving).

**Parallel axiom** can be demonstrated. While the Euclidean space accommodates exactly one parallel<sup>3</sup> to a line through a given point (not laying on this line), the H2 offers infinitely many. In  $PD$  a line is an arc whose ends are perpendicular to the circle rim. It is easy to draw many non-crossing circles through any given isolated point (see again Fig. 2).

**Exponential large space:** There exist two kinds of “parallels”: (i) *asymptotic parallels*, circles which touch at the rim in the same “ $\infty$ -point”, and (ii) *ultra parallels*, circle arcs which do not intersect within  $PD$ . One can anticipate, there is much more space in  $\infty$  than “usual” – sometimes the H2 is also called “more intensive infinite” than the  $\mathbb{R}^2$ . This extra space is desired for finding good accommodation for our data.

### 2.3 All in one Picture: Usage of Isometric Transformations

<sup>2</sup>Sometimes straight lines through the center point are referred to as *generalized circle* and considered as a circle arc with infinite radius.

<sup>3</sup>A *parallel* is by definition a line which does not intersect.

For changing the focus point in  $PD$  we need a translation operation (which can be associated to a mouse click or drag event). In general, the transformations which do not change form and metric in H2 are, similar to  $\mathbb{R}^2$  the translations, rotations, and the reflection. How do these operations look in the Poincaré  $PD$ ? Because all H2-lines remain H2-lines, their corresponding  $PD$  circle arcs must remain (generalized) circle arcs in  $PD$ .

The transformations that respect these properties (also called “circle auto-morphy” transformations) are here the Möbius transformations  $T(z)$  which can be formulated using complex numbers  $z, a, b$ :  $T_{a,b}(z) = (az + \bar{b}) / (bz + \bar{a})$  with  $|a|^2 - |b|^2 = 1$ . By describing the Poincaré disk  $PD$  as the complex unit circle, the isometric transformations for a point  $z \in PD$  can be rewritten in the more comprehensible form:

$$z' = T(z; c, \theta) = \frac{\theta z + c}{\bar{c}\theta z + 1}, \quad |\theta| = 1, \quad |c| < 1. \quad (3)$$

Here the complex number  $\theta$  describes the pure rotation of  $PD$  around the origin 0. The following translation by  $c$  maps the origin to  $c$  and  $-c$  becomes the new center 0 (if  $\theta = 1$ ). The inverse transformation is

$$T^{-1}(z; c, \theta) = T(z; -\bar{\theta}c, \bar{\theta}). \quad (4)$$

Two successive transformations in  $PD$  are computed faster by evaluating the concatenated transformation (see also [6]):

$$T(T(z; c_1, \theta_1); c_2, \theta_2) = T(z, c, \theta), \quad \text{with} \\ c = \frac{\theta_2 c_1 + c_2}{\theta_2 c_1 \bar{c}_2 + 1}, \quad \text{and} \quad \theta = \frac{\theta_1 \theta_2 + \theta_1 \bar{c}_1 c_2}{\theta_2 c_1 \bar{c}_2 + 1}$$

### 2.4 Data Accommodation in H2: Trees

Now we turn to the question raised earlier: how to accommodate data in the hyperbolic space. A solution to this question for the case of acyclic, tree-like graph data was provided by Lamping and Rao [7, 6]. By using mainly successive applications of transformation Eq. 3 they developed (and patented) a method to find a suitable layout for this data type in H2. Each tree node receives a certain open space pie segment, where the node chooses the locations of its siblings. For all its siblings  $i$  it calls recursively the layout-routine after applying the Möbius transformation Eq. 3 in order to center  $i$ .

Munzner developed an other graph layout algorithm for the three-dimensional hyperbolic space [9]. Her “H3Viewer” [10] allows fast drawing and frame rate interaction and arranges trees often as flower-like structures.

## 3. MULTIDIMENSIONAL SCALING (MDS)

First we review MDS and the Sammon algorithm.

Multidimensional scaling refers to a class of algorithms for finding a suitable representation of *proximity* relations of  $N$  objects by distances between points in a low dimensional – usually Euclidean – space. For a detailed analysis on proximity structures, see Hartigan [4]. In the following we represent proximity as *dissimilarity* values between pairs of objects, mathematically written as dissimilarity  $\delta_{ij} \in \mathbb{R}_0^+$  between the  $i$  and  $j$  item. As usual we assume symmetry, i.e.  $\delta_{ij} = \delta_{ji}$ . Often the raw dissimilarity distribution is not suitable for the low-dimensional embedding and an additional  $\delta$ -processing step is applied. We model it here as a monotonic transformation  $D(\cdot)$  of dissimilarities  $\delta_{ij}$  into

disparities  $D_{ij} = D(\delta_{ij})$ . In Sec. 6.3 we will discuss a transformation of adjusting a potential dimensionality mismatch.

The goal of the MDS algorithm is to find the spatial representation  $\mathbf{x}_i$  of each object  $i$  in the  $L$ -dimensional space, where the pair distances  $d_{ij} \equiv d(\mathbf{x}_i, \mathbf{x}_j)$  match the disparities  $D_{ij}$  as faithfully as possible  $\forall_{i \neq j} D_{ij} \approx d_{ij}$ . Here,  $L$  is typically two or three, since again the main purpose of MDS is visualization and explorative data analysis. The pair distance is usually measured by the Euclidian distance:

$$d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\| \quad \text{with } \mathbf{x}_i \in \mathbb{R}^L, i, j \in \{1, 2, \dots, N\} \quad (5)$$

### 3.1 Sammon's Algorithm

One of the most widely known MDS algorithms was introduced by Sammon [17] in 1969. He formulates a minimization problem of a cost function which sums over the squares of disparities–distance misfits, here written as

$$E(\{\mathbf{x}_i\}) = \sum_{i=1}^N \sum_{j>i} w_{ij} (d_{ij} - D_{ij})^2. \quad (6)$$

The factors  $w_{ij}$  are introduced to weight the disparities individually and also to normalize the cost function  $E$  to be independent to the absolute scale of the disparities  $D_{ij}$ . Depending on the given analysis task the factors can be chosen to weight all the disparities equally – the *global* variant ( $w_{ij}^{(g)} = \text{const}$ ) – or to emphasize the *local* structure by reducing the influence of larger disparities ( $w_{ij}^{(l)}$ )

$$w_{ij}^{(g)} = \frac{1}{\sum_{k=1}^N \sum_{l>k} D_{kl}^2}, \quad w_{ij}^{(l)} = \frac{2}{N(N-1) D_{ij}^2}. \quad (7)$$

Note that the latter is undefined if any pair has zero disparity. In his original work [17] Sammon suggested an *intermediate* normalization

$$w_{ij}^{(m)} = \frac{1}{\sum_{k=1}^N \sum_{l>k} D_{kl} D_{ij}} \quad (8)$$

which we are using in the following. Sammon proposed a steepest gradient method, in particular, the (diagonal) Newton method to iteratively minimize the remaining cost or stress  $E$ . He ignored the off-diagonal part of the Hessian matrix and used a step length reduced by a “magic” factor  $\eta$  of 0.3–0.4. Starting from a random  $\{\mathbf{x}_i\}$  initialization, in each iteration step, one object  $i^*$  is considered, and Eq. 6 minimized with respect to  $\mathbf{x}_{i^*}$

$$\mathbf{x}_{i^*}^{(new)} = \mathbf{x}_{i^*}^{(old)} + \eta \Delta_{i^*} \quad (9)$$

with  $\Delta_{i^*}$  here written per component  $q \in \{1, \dots, L\}$

$$\Delta_{i^*,q} = - \frac{\partial E}{\partial x_{i^*,q}} / \left| \frac{\partial^2 E}{\partial x_{i^*,q}^2} \right|. \quad (10)$$

The algorithm usually needs several epochs (with random sequence of selected  $i^*$ ) to converge to a cost-function minimum. The usual methods to deal with the risk of converging to a local minimum are restarting with different initial conditions and selecting the result with the lowest cost function or stress  $E$ . To save CPU time one may want to start with a good initial configuration  $\{\mathbf{x}_i\}$  for example by using the first  $L$  principal components found by PCA.

The reader is referred to [1] for further details on this and other MDS algorithms.

## 4. MULTIDIMENSIONAL SCALING IN H2

**The principal idea** is rather straight forward: instead of using the Euclidean  $L$ -dimensional  $\mathbb{R}^L$  as target space we apply Sammon's idea in the hyperbolic space  $PD$ . I.e. we replace the distance metric Eq. 5 by the appropriate distance metric for the Poincaré model (see, e.g. [8])

$$d_{ij} = 2 \operatorname{arctanh} \left( \frac{|\mathbf{x}_i - \mathbf{x}_j|}{|1 - \mathbf{x}_i \bar{\mathbf{x}}_j|} \right), \quad \mathbf{x}_i, \mathbf{x}_j \in PD. \quad (11)$$

Several aspects need consideration:

**Stay on disk:** Care must be taken to stay within the circle  $PD$  when updating the point  $i^*$ . Simple restriction of the update step  $\eta \Delta_{i^*}$  in Eq. 9 is not appropriate. Instead the Möbius transformation Eq. 3 leads in the right direction and handles the exponential shrinkage in the vicinity of the circle rim.

$$\mathbf{x}_{i^*}^{(new)} = T(\mathbf{x}_{i^*}^{(old)}; \eta \Delta_{i^*}, 1); \quad (12)$$

**Gradients:** While the gradients  $\partial d_{ij,q} / \partial x_{i,q}$  required in Eq. 10, are rather simple to compute for the Euclidean case ( $= (x_{i,q} - x_{j,q}) / d_{ij}$ ), the case becomes complex for Eq. 11

$$\begin{aligned} \frac{\partial}{\partial x_{i,1}} d(\mathbf{x}_i, \mathbf{x}_j) &= \frac{2t}{1-t^2} \left( \frac{v_1}{v_1^2 + v_2^2} - \frac{x_{j,1} v_3 + x_{j,2} v_4}{v_3^2 + v_4^2} \right) \\ \frac{\partial}{\partial x_{i,2}} d(\mathbf{x}_i, \mathbf{x}_j) &= \frac{2t}{1-t^2} \left( \frac{v_2}{v_1^2 + v_2^2} + \frac{x_{i,1} v_4 - x_{j,2} v_3}{v_3^2 + v_4^2} \right) \end{aligned}$$

with

$$\begin{aligned} \mathbf{x}_i &= (x_{i,1} + i x_{i,2}) \in PD \subset \mathbb{C} \\ \mathbf{x}_j &= (x_{j,1} + i x_{j,2}) \in PD \subset \mathbb{C} \\ v_1 &= x_{i,1} - x_{j,1} \\ v_3 &= x_{i,1} x_{j,1} + x_{i,2} x_{j,2} - 1 \\ v_2 &= x_{i,2} - x_{j,2} \\ v_4 &= x_{i,1} x_{j,2} - x_{j,1} x_{i,2} \\ t^2 &= \frac{v_1^2 + v_2^2}{v_3^2 + v_4^2} \end{aligned}$$

with two special cases: (i) the denominator  $v_1^2 + v_2^2$  becomes zero only if the points  $\mathbf{x}_i$  and  $\mathbf{x}_j$  are equal. (ii)  $v_3^2 + v_4^2 = 0$  does not occur for valid points inside the unit circle.

**Cost Minimization without Second Derivatives:** Due to the complexity of these results we do not compute second derivatives and rather improve the cost minimization step Eq. 10. We employed the Levenberg-Marquardt approach and followed the argumentation in [13] by dropping the higher derivatives. Each point  $i$  has an individual and bounded  $\lambda$ -parameter. More details will be reported elsewhere (in preparation).

**Disparity preprocessing:** Due to the non-linearity of Eq. 11 the preprocessing function  $D(\cdot)$  (see Sec. 3.1) has more influence in H2. Consider, e.g., linear rescaling of the dissimilarities  $D_{ij} = \alpha \delta_{ij}$ : in the Euclidean case the visual structure is not affected – only magnified by  $\alpha$ . In contrast in H2,  $\alpha$  scales the amount of curvature that is felt by the data. In the  $PD$  this has the effect that the increasing  $\alpha$  shifts all points further outside – but there is also much more (exponentially more) space. It depends on the given dataset and its dissimilarity structure, what is the optimal

$\alpha$ . In the experiments reported below, we set  $\alpha$  manually. We use this extra degree-of-freedom to choose a compromise between visibility of the entire structure and space for navigation in the detail-rich outer areas. It is certainly possible to integrate the optimization of  $\alpha$  in the overall minimization. However, for mapping problems with  $\mathbb{R}^2$  topology, this would obviously lead to zeroing  $\alpha$ , which is not intended.

## 5. FIRST EXPERIMENTS

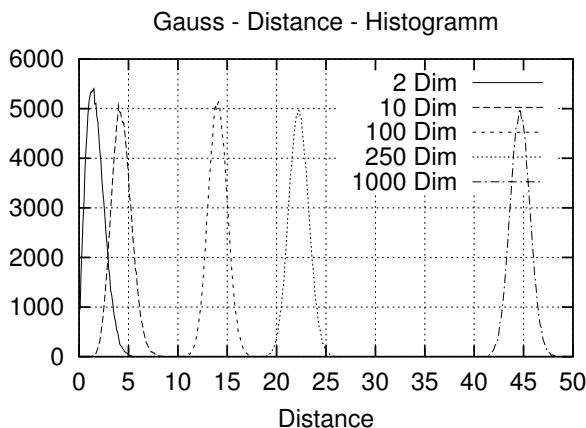
We conducted first experiments and present several real and synthetic datasets.

### 5.1 Iris Dataset

Fig. 5(a-d) displays the obligatory “Hello World” example of data mining – Fisher’s *Iris* data set. It describes 150 flower samples of three types: *iris setosa* (“ $\Delta$ ”), *iris versicolour* (“ $\times$ ”), and *iris virginica* (“+”).  $\delta$  is here the Euclidean pair distance in the four components sepal length, sepal width, petal length, and petal width. They demonstrate that MDS performs well in the hyperbolic plane and visualize the focus + context effect with three navigation snapshots.

### 5.2 Characteristics of High-Dimensional Distributions

Feature-rich datasets are inherently high-dimensional. With growing dimensionality most of their volume is contained in a thin, outer shell. This has consequences of the proximities distribution of those feature-rich, possibly heterogenous datasets. Using Monte-Carlo simulations we examined  $M$ -dimensional point distributions with (i)  $M$  i.i.d. Gauss random variables for each component, (ii) uniform distribution inside the unit sphere, (iii) on its surface, (iv) inside the  $M$ d-hypercube and (v) on its corners.



**Figure 4:** Histogram of pair distances  $\delta$  shown for various dimensions  $M$  (124,750 pairs from 500 Gauss distributed, random points with unit variance; binning width is 0.1).

Fig. 4 illustrates the first case. With growing  $M$ , the distance distribution  $\delta$  shifts to larger distances without significant widening. Even though the peak structures differ for smaller  $M$ , the observation holds also for the other cases

(ii-v) and can be generalized to inherently high-dimensional data, i.e. not laying in a low-dimensional sub-space.

As noted by several other authors [3, 5] the optimal embedding in low- $L$  spaces tends to a circular ring structure with growing  $M$ -dimensionality. Unfortunately, the situation bears numerous local minima and the Sammon algorithm often returns sub-optimal results (some available implementations exhibit numerical problems and return here bizarre structures, e.g. the *sompak*-package).

How does the H2 accommodate such a dataset? It turns out, that circumstances are much better since the cost function landscape offers more space to circumvent local minima. Fig. 5 shows the two target mappings for  $\mathbb{R}^2$  and H2 together with their disparity-distance scatterplots  $\{D_{ij}, d_{ij}\}$  ( $\alpha = 0.33$ ). The remaining stress was 40% higher for the Euclidean embedding with  $E_{R2} = 0.40$  versus  $E_{H2} = 0.285$ , which is comprehensible also in the two  $\{d_{ij}, D_{ij}\}$  scatterplots in Fig. 5(c,d).

## 6. APPLICATION TO NAVIGATION IN UNSTRUCTURED TEXT: THE “SPACE OF MOVIES”

In times of exponential growth of digital information the semantic navigation in datasets – particularly for the case of unstructured text documents – is a major challenge. In this experiment we demonstrate the application of the hyperbolic MDS to this situation.

### 6.1 Bag of Words – Standard Representation for Text Mining

In the domain of the information retrieval and text mining text is very often treated as *bag-of-words* and represented as very high-dimensional vector. One may argue that this ignores completely semantic information word order – amazingly the results justify this drastic step.

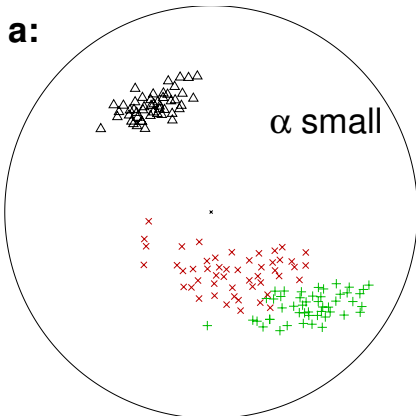
Given a collection of  $N$  text documents, first a vocabulary – a set of words  $\{w_i\}$  is determined. Words in text are always preprocessed by a suitable *word stemming* procedure and filtering out of *stop words*. The vocabulary is then the interesting part of all unique word stems, i.e. the most and the least frequent words are rejected. Each text document  $t$  is represented by a feature vector  $\vec{f}_t$ , where the components  $f_{t,i}$  are determined by

$$f_{t,i} = TF(t, w_i) \log \left( \frac{N}{DF(w_i)} \right). \quad (13)$$

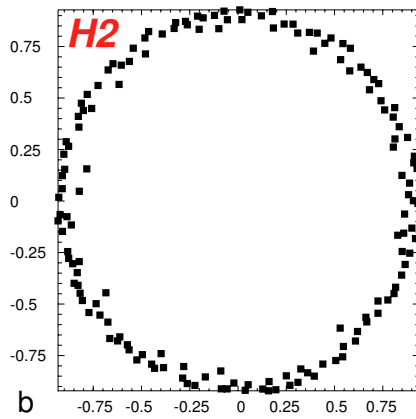
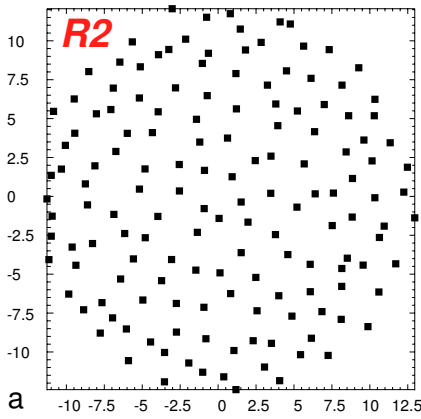
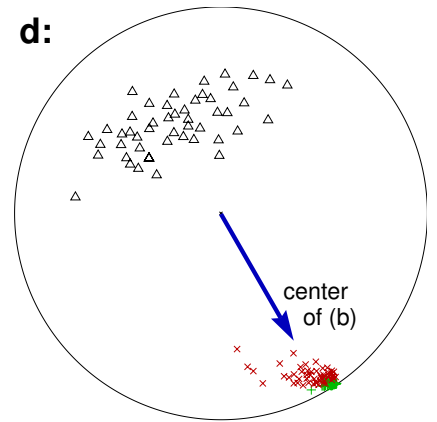
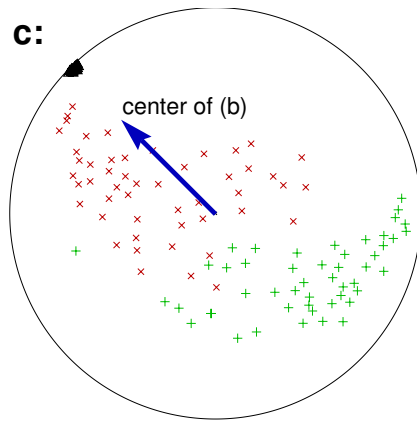
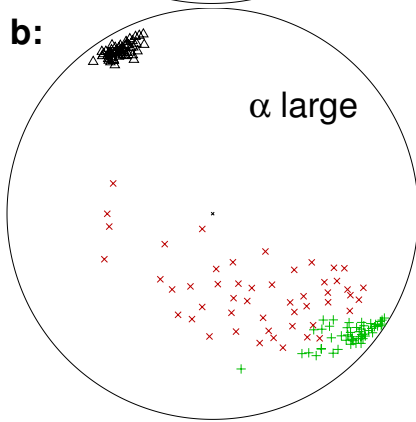
$TF(t, w_i)$  is the *term frequency* and counts the number of times the term  $w_i$  occurred in document  $t$ .  $DF(w_i)$  denoted the *document frequency* and counts the number of documents where the term occurred. This standard weighting scheme emphasizes rare words as more significant than common words, for more details see [16]. Proximity ( $= 1 - \delta$ ) and therewith dissimilarities of two documents is computed with the cosine metric

$$\delta_{ij} = 1 - \cos(\vec{f}_{t_i}, \vec{f}_{t_j}) = 1 - \frac{\vec{f}_{t_i} \cdot \vec{f}_{t_j}}{\|\vec{f}_{t_i}\| \|\vec{f}_{t_j}\|}, \quad \text{with } \vec{f}^j = \frac{\vec{f}^j}{\|\vec{f}^j\|} \quad (14)$$

and efficiently implemented by storing the normalized document feature vectors  $\vec{f}^j$ .

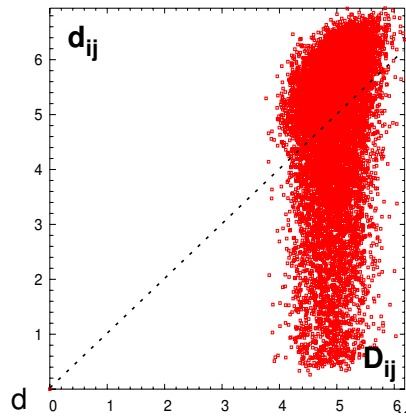
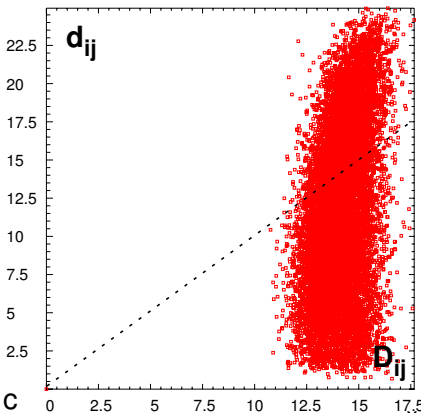


**Figure 3:** (a-d) Iris dataset: the three iris classes are clustered and well separated in PD as indicated by their markers. (a) uses a smaller  $\alpha$  for dissimilarity rescaling than (b). While (a) looks comparable to a common Sammon map in  $\mathbb{R}^2$ , in (b) – (d) the wider space is H2 is used and can be explored by moving the focus point (via mouse drag) to the areas of interest. The arrows indicate the original focus location in (b).



**Figure 5:** (Top) Multi-dimensional scaling of 150 Gauss distributed points in  $M = 100$  dimensions into the (a)  $\mathbb{R}^2$  and (b) into the hyperbolic H2.

(Top:) The MDS in Euclidian space (a) gets more easy locked in local minima than the MDS in the hyperbolic space. The right side (b) shows the ring-like structure previously reported to be more optimal for this high-dimensional mapping task. Probably the extra space in H2 allows to circumvent around the local minima during the iterative MDS process.



(Bottom:) The two corresponding scatterplot exhibit the target dissimilarities  $D_{ij}$  and the obtained distances  $d_{ij}$ . The optimal distribution has little rest stress and approximates the dashed diagonal line. (d) In H2 the majority of points lay closer than in the left diagram (c) which tends to smaller  $d_{ij}$  below the diagonal.

## 6.2 Computing the Movie Representation:

As an example we used a text collection consisting of film critiques taken from the rec.art.movies.reviews newsgroup augmented with genre information drawn from the internet movie database (<http://www.imdb.org>). The vocabulary with 5084 distinct terms was extracted and the feature vectors  $\vec{f}_t$  computed (for more details consult [11]). Based on this intermediate review database we selected all movies of the categories *animation* and *science-fiction* which had more than five recorded reviews. Then, each *average movie feature vector* was calculated from the set of its associated reviews

$$\vec{f}_m = \sum_{t \in A} \vec{f}_t, \quad A = \{t | t \text{ describing movie } m\}. \quad (15)$$

In contrast to the usage of directly accessible, brief movie plot description this procedure increases not only the amount of source information but also increases author independence by averaging.

Fig. 6 shows two hyperbolic MDS results of the 132 selected, well reviewed movies. The color markers indicate the genre of the accommodated movies. Even though this information was not encoded in the feature vectors, the hyperbolic MDS grouped the movies obviously in a semantically meaningful manner. The “+” marked film belong to both main genres displayed here – we find them in the middle, close to the separating border.

## 6.3 Proximity Contrast Enhancement:

Fig. 7 shows the histogram of the obtained 8646 dissimilarities for the 132 selected, well reviewed movies. The distribution resembles a high-dimension random distribution, but with an “effective” dimensionality (compare Fig. 4) certainly lower than the 5084 dimensions of the movie feature vector  $\vec{f}$ .

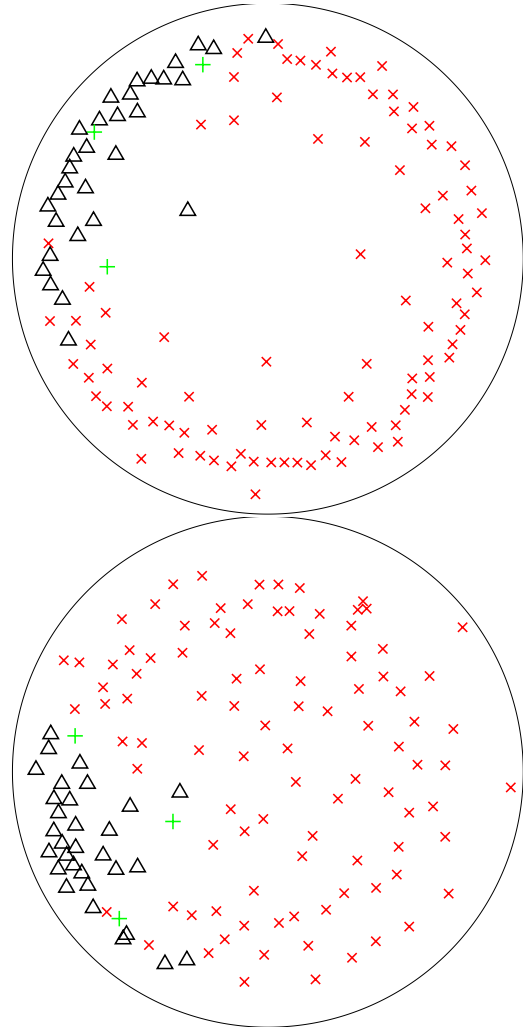
Following an idea from [5] we experimented with non-linear transformation  $D(\cdot)$ , which shifts effectively the high-dimensional dissimilarity modes, seen in Fig. 4, to the left. The mathematical details are reported elsewhere (in preparation). The first general effect is the reduction of the “ring-like” structure toward a more uniform movie placement as seen in the left versus the right graphic in Fig. 6. The main rationale is the *contrast enhancement* between documents or movies: we want to see related movies closer than their pure review text statistics would justify.

The improvement can be examined in the lower part of Fig. 6. The lower left group of triangle, indicating the genre *animations*, displays stronger cohesion. In conjunction with the fully labeled snapshots in Figs. 8,9 they can be identified as Disney World cartoons. Furthermore two tight groups appear: the lower left  $\Delta$  pair represent “Toy Story 1+2”, the  $\times$ -cluster on the opposite (1 o'clock) can be identified as four “Alien” films (see Fig. 9).

## 7. DISCUSSION AND CONCLUSION

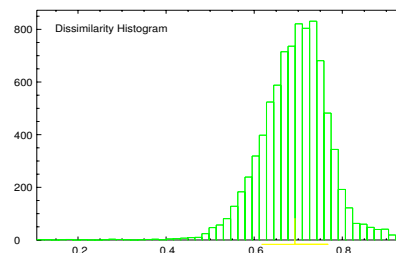
The snapshots in Figs. 8,9 give a glimpse at the potential exploratory usage of the introduced H-MDS, the hyperbolic multi-dimensional scaling technique.

Obvious clusters like the “Star Trek” group, or the proximity of “Bug’s Life” and “Antz” strike. Cineastes may want to mine further examples in the appendix figures. Note, that no extra information was imposed. The spatial struc-



**Figure 6: Genre separation with and without distance contrast enhancement.** The movie become well separated along their major genres: the red “ $\times$ ” mark *science-fiction*, the black “ $\Delta$ ” *animation*, and the green “+” mark films belonging to both genres. Again, this genre information was marked in post processing – not given.

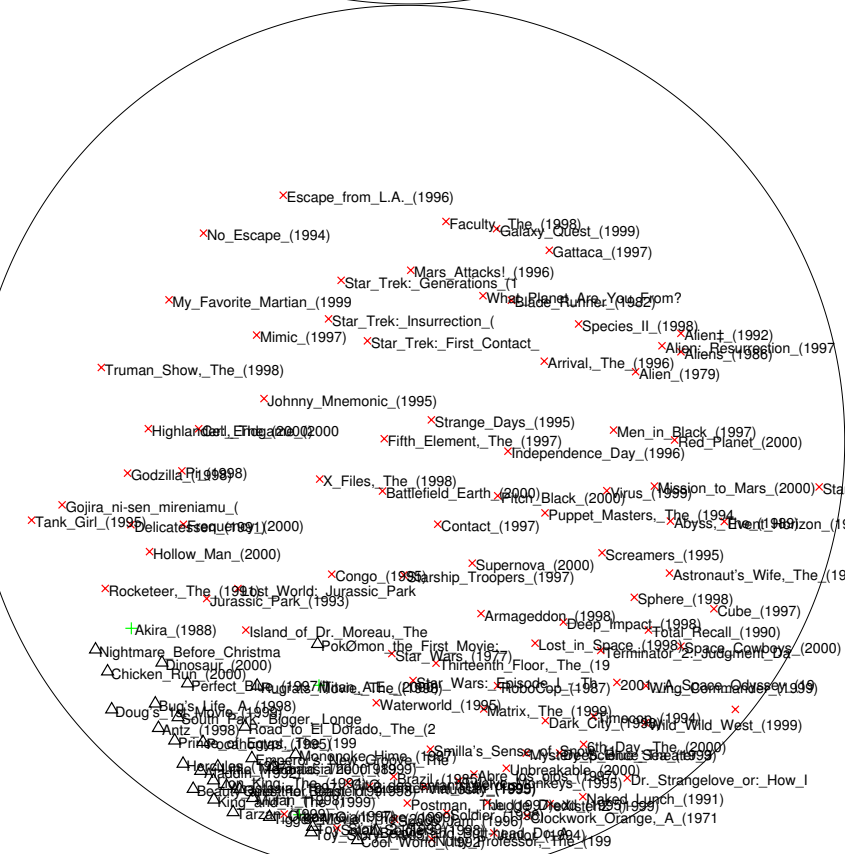
(*Top:*) Without the distance contrast enhancement a ring structure is visible; (*bottom*) turning it on, allows to better populate the center area and strongly related films can get now very close in *PD*. See the  $\Delta$ -couple (7 o'clock) and the  $\times$ -cluster (1 o'clock).



**Figure 7: Histogram of dissimilarities  $\delta_{ij}$  for the selected movies.**



**Figure 8: Four navigation snapshots (cont. in next Fig. 9) through the developed “space of movies”. (Top:) The mapping is equal to the right side of Fig. 6 but with labeling turned on. (Bottom:) By shifting the focus to the upper science-fiction region (‘x’ marked) one can discover, e.g., a band of “Star Trek” movies, around 11 o’clock. Now the tight “Alien” cluster can be inspected: at 2 o’clock four movies of this series gathered very close.**





turing was automatically determined, driven by the (dis-)similarities of word frequencies in secondary, descriptive texts of totally unrelated authors.

These results show that the H-MDS approach is well suited to support “semantic browsing” in datasets or document collections. Compared to the HSOM approach with a fixed grid [11], the H-MDS freely builds a spatial structure and can therefore display the closeness of objects – as seen, e.g., in the tight “Alien” cluster (Fig. 6). Through the *proximity transformation* the resulting distribution and the amount of visible information can be adjusted to the desired situation.

From MDS the H-MDS inherits the limiting scaling behavior of  $N(N - 1)/2$  required dissimilarities for  $N$  points. On the positive side it also allows to process non-vectorial data, available only as pairwise proximities.

From the embedding in H2 this new approach profits not only from the extra space for compressing semantic relationships. It also gains the superb visualization and navigation properties, which were found to yield significant improvement in task time compared to traditional browsing methods.

Future work will address (i) the automatic customization of the proximity transformation, (ii) the combination of the H-MDS with the HSOM for visualizing very large datasets. (iii) When looking at the label intense appendix figures, it appears obvious that the display can be successfully combined with various search operations and suitable graphical attribution.

## 8. REFERENCES

- [1] Trevor F. Cox and Micheal A. Cox. *Multidimensional Scaling*. Monographs on Statistics and Applied Probability. Chapman and Hall, 1994.
- [2] H.S.M. Coxeter. *Non-Euclidean Geometry*. University of Toronto Press, 1957.
- [3] J. deLeeuw and I. Stoop. An upper bound for SSTRESS. *Psychometrika*, 51:149–153, 1986.
- [4] J.A. Hartigan. Representation of similarity matrices by trees. *J. Am. Statist. Ass.*, 62:1140–1158, 1967.
- [5] Hansjörg Klock and Joachim Buhmann. Multidimensional scaling by deterministic annealing. In *Proc EMMCVPR Venice*, 1997.
- [6] J. Lamping, R. Rao, and P. Pirolli. A focus+context technique based on hyperbolic geometry for viewing large hierarchies. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 401–408, 1995.
- [7] John Lamping and Ramana Rao. Laying out and visualizing large trees using a hyperbolic space. In *ACM Symposium on User Interface Software and Technology*, pages 13–14, 1994.
- [8] Frank Morgan. *Riemannian Geometry: A Beginner's Guide*. Jones and Bartlett Publishers, 1993.
- [9] Tamara Munzner. H3: Laying out large directed graphs in 3d hyperbolic space. In *Proceedings of the 1997 IEEE Symposium on Information Visualization, Phoenix, AZ*, pages 2–10, 1997.
- [10] Tamara Munzner. Drawing large graphs with h3viewer and site manager. In *Proceedings of Graph Drawing '98, Montreal, Canada, Springer-Verlag*, Lecture Notes in Computer Science 1547, pages 384–393, 1998.
- [11] Jörg Ontrup and Helge Ritter. Text categorization and semantic browsing with self-organizing maps on non-euclidean spaces. In *Proc PKDD-2001*, pages 338–349. Springer LNAI 2168, 2001.
- [12] Peter Pirolli, Stuart K. Card, and Mija M. Van Der Wege. Visual information foraging in a focus + context visualization. In *CHI*, pages 506–513, 2001.
- [13] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C – the Art of Scientific Computing*. Cambridge Univ. Press, 1988.
- [14] Kirsten Ridsen, Mary P. Czerwinski, Tamara Munzner, and Daniel B. Cook. An initial examination of ease of use for 2d and 3d information visualizations of web content. *International Journal of Human Computer Studies*, 53(5):695–714, 2000.
- [15] H. Ritter. Self-organizing maps on non-euclidean spaces. In S. Oja, E. & Kaski, editor, *Kohonen Maps*, pages 97–110. Elsevier, Amsterdam, 1999.
- [16] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 5(24):513–523, 1988.
- [17] J. W. Sammon, Jr. A non-linear mapping for data structure analysis. *IEEE Transactions on Computers*, 18:401–409, 1969.
- [18] J.A. Thorpe. *Elementary Topics in Differential Geometry*. Springer Verlag, 1979.
- [19] William P. Thurston and Jeffrey R. Weeks. The mathematics of three-dimensional manifolds. *Scientific American*, July:94–107, 1984.

## Acknowledgment

Daniel Weßling deserves many thanks for his help in conducting experiments and improving code and Jörg Ontrup for sharing his movie expertise and review database. Thanks to Jeffrey Weeks for Fig. 1 and to Cordon Art - Baarn, Holland for permission to use Fig. 2 (all rights reserved on all M.C. Escher works © 2002).