DOI: 10.1111/cgf.15113
EUROVIS 2024
C. Garth, A. Kerren, and R. Raidou
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 43 (2024), Number 3
STAR – State of The Art Report

State of the Art of Graph Visualization in non-Euclidean Spaces

Jacob Miller[1], Dhruv Bhatia[1] and Stephen Kobourov[1]

[1]University of Arizona, Tucson, USA

Abstract
Visualizing graphs and networks in non-Euclidean space can have benefits such as natural focus+context in hyperbolic space and the familiarity of interactions in spherical space. Despite work on these topics going back to the mid 1990s, there is no survey, or a part of a survey for this area of research. In this paper we review and categorize over 60 relevant papers and analyze them by geometry, (e.g., spherical, hyperbolic, torus), by contribution (e.g., technique, evaluation, proof, application), and by graph class (e.g., tree, planar, complex).

1. Introduction

When one begins to visualize a graph or network using a node-link diagram, one typically imagines placing the nodes and routing the edges in 2 dimensional Euclidean space; the space of the computer screen or a sheet of paper. However, representations in different geometries have distinct visualization benefits, such as natural focus+context [LRP95, MB95], or the possibility to obtain a more faithful representation with respect to quality metrics [KPK*10, SSGR18].

Several series of papers on non-Euclidean graph visualization appeared near the turn of the last century and recently this subject is of interest again to visualization researchers. While there are many surveys on graph visualization in general, few consider visualization in non-Euclidean spaces and there are none that cover all of spherical, hyperbolic and toroidal graph visualizations. There are several graph visualization surveys for specific types of graph data, such as dynamic networks [BBDW17], multi-layer networks [MGM*19], and multi-variate networks [NMSL19]. There are also more general surveys on graph visualization with topics on group structures [VBW17], and scientific visualization [WT17]. Von Landesberger et al. [VLKS*11] survey visualizations of large networks, in which they point out that hyperbolic layouts scale well to large trees, though the survey itself does not overlap in content with ours.

We model our methodology on similar surveys. Kale et al. review dynamic multivariate networks [KSP23] and use a similar multi-dimensional tagging system to our methodology. A related survey categorizes works in geospatial network visualization [SYPB21]. While geospatial networks are (implicitly or explicitly) embedded on a sphere, these positions are typically given and have a very specific geographic interpretation. We include graph visualization techniques that put arbitrary graphs on the sphere (not necessarily geographic), as well as in other geometric spaces. Chen [CHE22] describes a systematic exploration of the design space for wrappable visualizations, which includes spherical and toroidal graph visualizations, but not hyperbolic visualizations.

1.1. Motivation

Graph visualization beyond the standard 2 or 3 dimensional Euclidean space has been a topic of interest dating back to the 1990s. Spherical [GSF97, WT06], hyperbolic [Mun97, LRP95], and toroidal [CDMB20] geometries have been considered. Some user-studies have also shown benefits to using these geometries [DCL*17, CDBM21]. Applications of non-Euclidean geometry in network visualization include the sphere [LC10, KMLM15, BKP*12], the hyperbolic plane [Mun00, Gla19], and the torus [MIB*14, CZIM18]. There are also several tools that include some non-Euclidean graph components [LRP95, BM12, LD12]. More recently, non-Euclidean representations of graphs and other data have become a topic of interest for neural networks [PVM*21] and dimensionality reduction [GGY22]. In this survey we review the state of the art in graph visualization in non-Euclidean spaces.

1.2. Contribution

In this paper we review relevant papers in graph visualization in non-Euclidean spaces and then analyze the state of the art by geometry, (spherical, hyperbolic, torus), by contribution (technique, evaluation, proof, application), and by graph class (e.g., trees, small-world graphs, large networks).

While reviewing papers, we aimed to identify gaps in the literature and current open problems that may be of interest to researchers. These problems are mentioned as relevant throughout the paper and are summarized in Section 8. We additionally hope this survey serves as a reference for interested researchers, by providing the basics of non-Euclidean geometry, a bit of historical background, and a review of the current research.

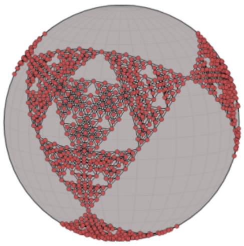
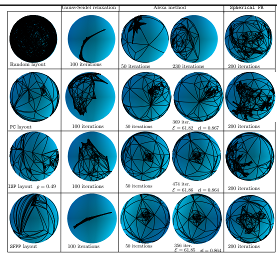
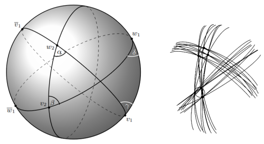
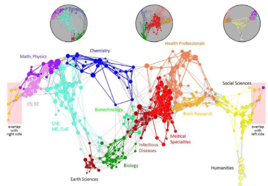
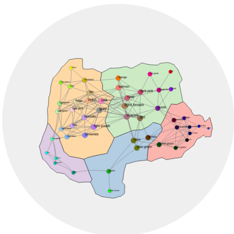
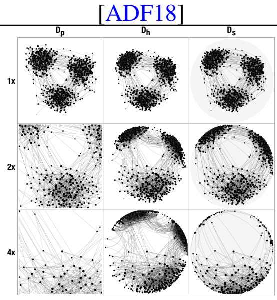
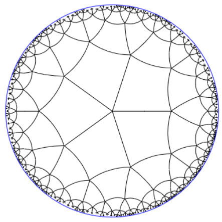
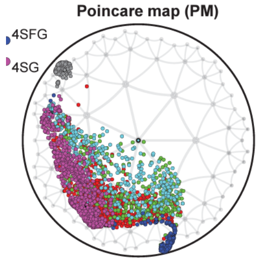
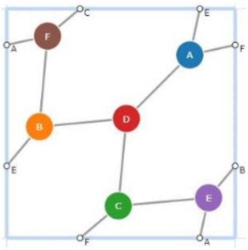
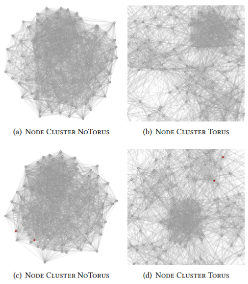
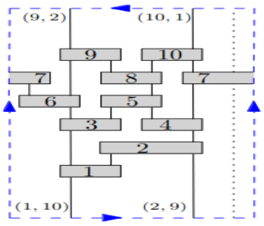
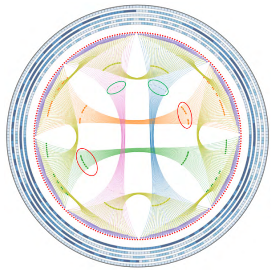
	Technique	Evaluation	Proof	Application
Spherical	 [MHK23]	 [ADF18]	 [BMW20]	 [BKP*12]
Hyperbolic	 [MKH22]	 [DCL*17]	 [Epp21] via [Hat]	 [ZS21]
Toroidal	 [CDMB20]	 [CDBM21]	 [Bie22]	 [CZIM18]

Table 1: *Example works in a given geometry (row) and type of contribution (column).*

2. Methodology

Here we briefly summarize our paper selection criteria and our paper classification approach.

2.1. Selection Criteria

We focus our attention on node-link diagram representations in non-Euclidean (or Riemannian) spaces. We initially generated a candidate set of papers via a DBLP API query, which includes the proceedings of *IEEE VIS, IEEE PacificVis, EuroVis, CHI*, and the *Symposium on Graph Drawing*, along with the journals *TVCG* and *CGF*. Our exact search query was "graph|network+visual|draw|layout+hyperbol|spher|riemman|torus|non-Euclidean", and returned an initial corpus of 43 papers. We supplemented this with prior knowledge of the field, which yielded 13 additional papers that did not come up in the DBLP query. While reviewing these papers, we paid close attention to the related work cited in them, and added the relevant citations. This yielded an additional 8 papers. Another 2 relevant papers were added from suggestions by the reviewers of this survey, bringing the total of non-Euclidean graph papers reviewed to 66.

2.2. Classification

We consider three dimensions in our analysis of the related work, described below along with our technique for collecting papers.

What is the geometry? We make note of the specific geometry studied for the graph visualization in the paper. We identify three non-standard geometries that are used for graph visualization:

- **Spherical geometry** is the surface of a 3-dimensional sphere, used to depict graphs with geographic or geographic-like information.
- **Hyperbolic geometry** is analogous to the sphere with 'opposite' curvature. Most notable for its exponentially expanding space allowing for 'perfect' drawings of trees and hierarchies.
- **Toroidal geometry** is the surface of a 3-dimensional ring or 'doughnut'. Admits drawings of some non-planar graphs without crossings and can be embedded in the plane with no distortion.

What is the contribution? Our second dimension of analysis indicates the type of contribution a paper makes. We identify four type of contributions:

- **Technique** denotes papers whose main contribution is layout algorithm and drawing style.

- **Evaluation** are papers which include some algorithmic or human-subjects evaluation of a technique's effectiveness.
- **Proof** style papers are theoretical papers that present results about embedding graphs in non-Euclidean space.
- **Application** papers provide a specific application in non-Euclidean geometry, such as showing how a given dataset can be analyzed.

What types of graphs? The third dimension of our analysis is the type (or class) of graphs a paper considers in its scope. These include strict definitions such as trees and planar graphs, as well as more fuzzy definitions such as complex networks.

In summary, we review work on graph visualization in non-Euclidean spaces and provide a taxonomy and analysis. We believe this survey is timely, with recent interest on the subject in the visualization and human-computer interaction communities, and will provide interesting directions for future research where much has yet to be done.

3. Terminology and Background

We briefly introduce terminology used throughout the paper, then give a background on non-Euclidean geometry.

3.1. Terminology

Topological graph theory studies embedding of graphs on surfaces [GT01]. Graphs have been used to model real world problems, dating back to as early as 1736 and Leonhard Euler [Eul41]. The terminology has changed and many variations exist (e.g., node/vertex, edge/link/arc), but here we aim for consistency.

A ***graph*** (network) is a finite set of objects V (called *vertices*) together with a set of relationships on those objects $E \subseteq V \times V$ (called *edges*). Edges may be directed or undirected, and potentially weighted. Graphs arise in many application areas, making their visualization of particular research interest. While some fields make a distinction between graphs and networks, we treat them interchangeably in this survey.

A common visualization idiom for graphs is the ***node-link diagram***, with vertices drawn as nodes (simple shapes) and edges drawn as links (curves between the corresponding vertices). Node-link diagrams are constructed from an ***embedding*** (layout) of the graph, an assignment of positions to the vertices and a routing of edges. The resulting picture is called a ***drawing*** of the graph. In a typical straight-line drawing, the routing of the edges is totally determined by the placement of the incident vertices, as the line segment between them.

There are many classes of graphs that are subject of particular interest. ***Trees*** are graphs that contain no cycles. If a root of the tree is specified, this encodes a hierarchy, where each non-root vertex has exactly one parent it descends from. ***Planar graphs*** are graphs which can be drawn as a node-link diagram without any two links intersecting. ***Complex networks*** are graphs that arise naturally in real-world networks, defined by phenomena such as small average path length and high clustering coefficient.

Many dimension reduction (DR) techniques can be applied to obtain a graph embedding. These techniques take as input a distance matrix, D, on a set of high dimensional Euclidean coordinates where $D_{i,j}$ is the distance between objects i and j. Defining distance with the graph theoretic (i.e. shortest path) distance allows one to apply these DR techniques to graph embedding. A particular technique used in graph embedding is ***multi-dimensional scaling (MDS)*** (stress minimization), which tries to match the pairwise embedding distances to the original input distances. Examples of applying DR methods to graph layout include stress majorization [GKN04], PivotMDS [BP06], and t-SNET [KRM*17].

In the plane, one can draw a straight line segment between two points and measure its length – this is known as the Euclidean distance or L_2 norm. This intuitive concept can be generalized to arbitrary surfaces with the ***geodesic***: the length of the shortest curve between two points. In non-Euclidean geometries (such as 2D sphere and 2D hyperbolic space), this is a circular arc defined by a closed-form function. In Euclidean and hyperbolic space, there is a unique 'straight line' between any two points. For the sphere there may be many possible straight lines if the points are antipodal, but the length is the same for each. On the torus, however, there are several possible lines so it is required to measure the length of each of them to compute the geodesic length.

Non-Eulcidean geometries cannot be perfectly embedded into the 2 dimensional plane. The study of cartography is a field dedicated to this problem [Sny87] as it has long been known for the sphere. A ***projection*** is a linear map that preserves some combination of angles, areas, geodesics, or distances but a projection (from spherical or hyperbolic space) that preserves all these aspects simultaneously is not possible.

3.2. Background on non-Euclidean and Riemmanian Geometry

Geometry is among the oldest and most studied areas of mathematics, with roots in ancient times. Around 300 BC, Euclid wrote his *Elements* which employed the notion of an axiomatic system: a mathematical system in which all statements should be proved from a small number of indisputably true axioms (postulates). Euclid gave five axioms which make up what we now call Euclidean geometry.

1. Any two points describe a (unique) line.
2. Any line segment can be extended to a line.
3. A circle is described by a center and radius.
4. All right angles are equivalent.
5. "That, if a straight line falling on two straight lines make the interior angles on the same side less than two right angles, the two straight lines, if produced indefinitely, meet on that side on which the angles are less than two right angles."

The fifth postulate (which states that any two non-parallel lines will intersect exactly once) is noticeably more involved than the previous four. Mathematicians tried for centuries to prove the fifth postulate using the first four, but invariably failed. Despite its perceived clunky-ness, the fifth postulate seemed necessary to prove even simple theorems such as the Pythagorean theorem about right angle triangles.

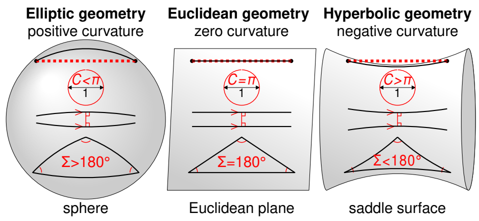


Figure 1: *Illustration of non-Euclidean geometries and Playfair's axiom (middle two lines). Image credit* `https://commons.wikimedia.org/w/index.php?curid=94781281`

3.2.1. Hyperbolic geometry

As it would turn out, the Euclidean notion of parallel lines is not necessary for a consistent geometric system. It was independently discovered by mathematicians Nikolai Lobachevsky and János Bolyai around 1830, that by allowing arbitrarily many parallel lines to exist, a distinct geometry arises. This non-Euclidean geometry came to be known as *hyperbolic geometry*.

The fifth postulate is equivalent to the following statement, known as Playfair's axiom: "Given a line, L, and a point, p, not on L, ***exactly one*** line parallel to L can be drawn through p"; see Fig 1 for an illustration. Then, hyperbolic space is obtained by replacing the fifth postulate by the statement: "Given a line, L, and a point, p, not on L, ***infinitely many*** lines parallel to L can be drawn through p."

Most properties of Euclidean geometry hold in hyperbolic geometry; for instance, any two lines can intersect at most once. Things differ when parallel lines or more than two lines are involved. Hyperbolic triangles have many properties distinct from traditional trigonometry, notably that the sum of their internal angles is strictly less than 180; See Figure 1.

Another notable difference is the absence of relative scale. In Euclidean geometry, affine transformations are linear maps of the plane to itself that can be nicely represented as a function with real valued matrix A and vector b, $f(x) = Ax + b$. This includes translations, reflections, rotations, and scaling (resizing). Once we have a drawing, we can shrink it to a tablet screen or expand it to a billboard and shapes (i.e. angles) are unaltered. In hyperbolic space however, there is a relationship between distance and angles. Translations, reflections, and rotations can all preserve angles, but scaling will not.

3.2.2. Spherical (elliptical) geometry

Although the geometry of the surface of a sphere had been studied for many centuries (e.g., work by Theodosius around 200 B.C. [Ros12]), it was considered a separate system with its own rules. Even today, the sphere is often thought of as a surface embedded in 3 dimensions of Euclidean space, but its geometry can be studied intrinsically only using the 2 dimensions of its surface.

It wasn't until 1859 that the classical understanding of spherical geometry was unified with the new ideas produced from the study of non-Euclidean geometry [Cay59]. Cayley introduced *elliptic* geometry, a sphere with antipodal points identified to satisfy the result (present in both Euclidean and hyperbolic) that lines can intersect at most once (note that on a full sphere, lines always intersect twice). Then elliptic geometry is obtained by the following modification of Playfair's axiom: "Given a line, L, and a point, p, not on L, ***no*** lines parallel to L can be drawn through p." Or in other words, all lines must intersect.

Although elliptic geometry removes the trouble of antipodal points not satisfying the first axiom, we generally refer to spherical geometry (also called doubly elliptic geometry) throughout this paper, as it is more widely used for visualization.

While the first, second, and fourth of Euclid's postulates hold in spherical geometry, the third must be slightly modified – a circle of arbitrary radius cannot exist. Since the sphere is finite, a circle can only have radius at most equal to the radius of the sphere it exists on.

Many of the unique properties of hyperbolic geometry have the opposite implication for spherical geometry. For instance, the sum of internal angles of a spherical triangle is strictly greater than 180 as seen in Fig. 2. Spherical space also has an absolute scale though since the surface is finite there is a strict upper limit on how much things can be stretched.

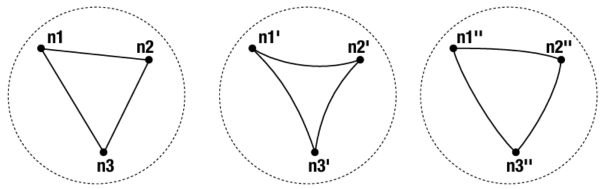


Figure 2: *Comparison of visualization of a triangle in Euclidean plane, hyperbolic plane, and sphere [DCL*17]. Note how the hyperbolic triangle has smaller angles while the spherical triangle has larger.*

3.2.3. Riemannian and Toroidal geometry

Riemannian geometry is the study of manifolds (spaces which locally resemble Euclidean space) together with a Riemannian metric (an inner product on the tangent space at each point). This includes the Euclidean and non-Euclidean geometries with the standard dot product, along with many other surfaces. Of these, only the torus has been used for graph visualization and is worth exploring.

A torus is a product of two circles that is topologically closed, which produces the typically doughnut-shape. The torus can be described as the surface of an axis of revolution of a circle in 3 dimensional space. It is an example of a surface with genus 1, where genus is the maximum number of closed, circular cuts that can be made and the surface remain connected. Note that the spaces discussed before all have genus 0.

A property of interest in graph drawing is the crossing number of the graph [Sch12], which is the minimum number of crossings required to draw a graph on a 2 dimensional surface of genus 0. A surface with higher genus admits smaller crossing numbers [Hea90]. Notably, the two cannonical non-planar graphs (K_5 and $K_{3,3}$) can be realized without crossings on the torus.

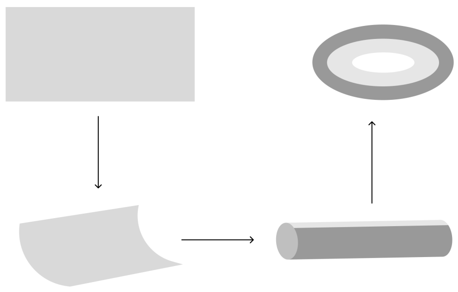


Figure 3: *Creating a torus from a piece of paper. Note how first two opposite ends of the rectangle are connected, then the top/bottom of the cylinder is connected to create the torus. This type of torus is generally called a flat torus. The light and dark grey indicate the inside and outside of the paper respectively.*

Although the surface of a torus is often depicted in 3 dimensions with curvature, a flat torus actually has curvature of 0 (the Euclidean metric), just like the Euclidean plane. Similar to how one can bend a sheet of paper into a cylinder in 3 dimensions with no distortion, the torus could be created from a sheet of paper with no distortions if we had a 4th dimension to work with. While no 3 dimensional embedding without distortion is possible, a nearly perfect embedding of the flat torus is obtained in 2 dimensions by a rectangle with the left/right and top/bottom edges identified as displayed in Fig. 3.

4. Geometry of Visualization

Our first dimension of analysis is geometry. We give a brief overview of the chronology of how each geometry has been used in graph visualization.

4.1. Sphere

Spherical geometry has been used in visualization since antiquity. Much of the visualization work on the sphere has been inspired from the Earth's globe, with orthographic projections and map metaphors being common.

Gross et al. [GSF97] describe an approach for graph layout on the sphere. This is a force-directed method which places nodes initially with an initialization scheme that attempts to keep nodes close to the surface of the sphere. Edges are not routed along geodesics, but as straight-line segments in 3D Euclidean space.

Sangole and Knopf [SK03] apply a self-organizing map (SOM) that creates closed spherical surfaces from the input data, transforming the discrete input into a continuous distribution on the sphere. For a globe-like feel, they doubly encode the density of the input data with a rainbow color map and the height of the point on the globe. Low density areas are blue and the lowest points to resemble oceans, while high density areas are red and high, resembling mountains.

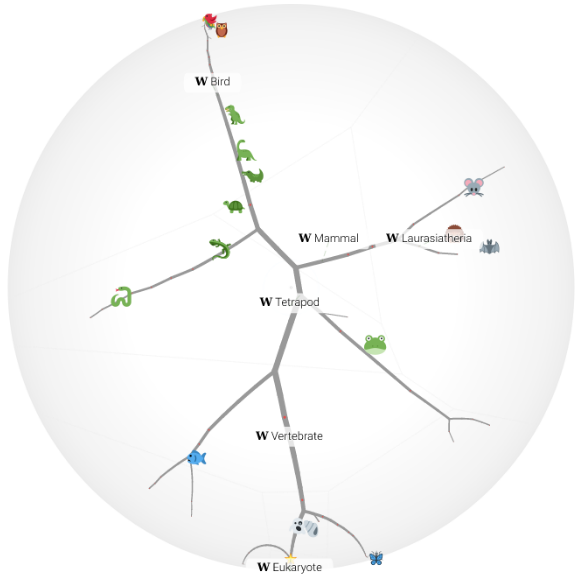


Figure 4: *Hyperbolic Tree of Life for browser based hyperbolic visualization. Source code and demo found from [Gla19]*

Wu and Takatsuka [WT06] adapt a SOM to the sphere for multivariate network visualization. This requires adapting the grid used in a SOM to a triangulation of the sphere. They also implement a visual interface to view the embedding with either an orthographic or cylindracal projection. The spherical SOM has also been used in the field of networking [TKY11], to visualize file sharing activities on a computer network with the help of an orthographic projection.

Traditional force-directed algorithms were generalized to the sphere [KW05] and the tree visualization scheme SphereTree was created in [BM12]. MDS has also been generalized to the sphere with a few different approaches. Osinska et al. [OB08] perform 3D Euclidean MDS and normalize the vectors to lie on a unit sphere. Another approach is to perform Euclidean MDS in 3D, but constrain the optimization so that the points never leave the surface of a sphere during the algorithm [DLM09, PYGK20]. Finally, [MHK23, EKK05] describe methods to perform MDS using latitude and longitude coordinates, constraining the computations and movements to the surface of the sphere.

Spherical layouts have been investigated in an immersive setting such as virtual reality [KMLM16, YJD*18], discussed further in section 5.4. The Map of Science is an application of spherical network visualization [BKP*12]; See Table 1 Spherical-Application. Other examples include the “Places and Spaces" [Bö12] and “Worldprocessor" [Gü07] exhibitions.

4.2. Hyperbolic

Hyperbolic space first received attention in the visualization community due to its exponentially expanding structure, and eye catching representations, in particular the M.C. Escher works inspired from the Poincaré disk [EV89], a model of hyperbolic space which maps lines to circular arcs in Euclidean space and admits many aesthetically pleasing tessellations. Hyperbolic embeddings are also of interest in the field of machine learning, as a desirable latent space for embeddings of neural networks [PVM*21] or embedding of general data [GGY22]. Here we cover works which deal primarily with hyperbolic geometry in graph visualization.

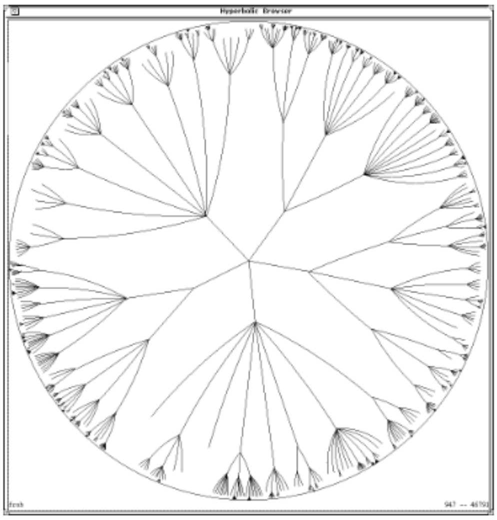
One of the earliest approaches of graph drawing in the hyperbolic plane is by Lamping et al [LRP95], dubbed the hyperbolic browser; see Fig. 5(a). They embed trees in hyperbolic space using a linear time algorithm and construct the embedding top-down from parent to children, evenly spacing children on the half-plane opposite the parent all at unit length. This is always possible thanks to the exponentially expanding hyperbolic geometry. To represent this embedding, they use the Poincaré disk which maps the entirety of hyperbolic space into a Euclidean disk. Since the entire tree is visible (up to pixel resolution) this provides a powerful focus+context effect, allowing one to see the parent node and its children in high detail, but also get a sense of how deep or wide each of the subtrees are. Clicking on a node places it in the center of the projection, via a smooth transition in the drawing. This approach was generalized to arbitrary graphs by [HHDK00] by first computing a spanning tree and filling in the removed edges after layout (or keeping them hidden).

Several application papers made use of the Lamping et al. [LRP95] technique. Phylogenetic trees [BS00], word trees [BW02], the tree of life [Gla19] all made natural candidates, as large trees with interest to domain experts. The hyperbolic browsing system also made its way into several systems and tools, such as the Hierarchy Visualization System (HVS) [APN07], the Java InfoVis Toolkit [Bel], and TreeBolic [Bou].

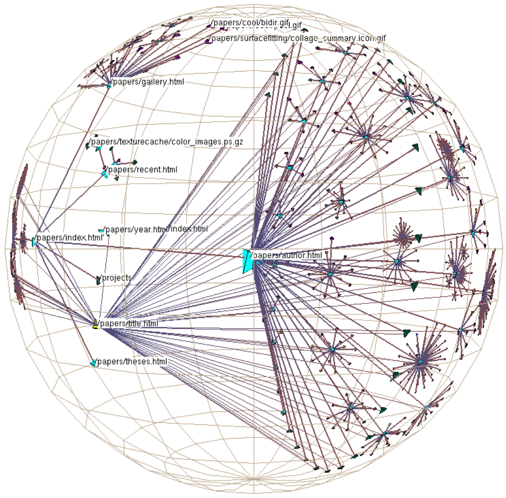
More recently, Lamping et al.'s [LRP95] hyperbolic browser has been re-implemented using d3.js [Gla18,Gla] for web-based visualization of large hierarchies such as the tree of life [Gla19] as shown in Fig. 4.

General graphs can be drawn in hyperbolic space as well, Miller et al. [MKH22] provide several different techniques to do so, building on earlier approaches from [KW05, Wal04, PYGK20].

While most work considers the 2-dimensional hyperbolic plane, Munzner uses 3D hyperbolic space to visualize trees [Mun97, Mun98, MB95], one of the examples shown in Fig. 5(b). Just like 2 dimensional hyperbolic space can be projected into the 2 dimensional plane, 3 dimensional hyperbolic space can be projected into the 3 dimensional Euclidean space. While the Poincaré projection is preferred in two dimensions due to its simple construction and conformal properties, it maps geodesics to curved arcs which are more difficult for a reader to follow in 3 dimensions. For this reason, Munzner chose to use the Beltrami-Klein projection in spite of the larger distortion, as it maps geodesics to Euclidean lines. The ball itself can be rotated when viewed from outside, or entered by zooming to get a closer look. The node in the center of the ball can be changed by navigating the tree. This technique was later extended to general graphs by first computing a spanning tree [Mun00], laying out with the above technique, and filling in the cycles afterwards. Munzner's work has been re-implemented in two subsequent systems: Walrus [Hyu00] and h3py [ZK16].



(a) Hyperbolic Browser from [LRP95]



(b) H3 from [Mun98]

Figure 5: *Hyperbolic tree visualizations either took the 2 dimensional (a) or 3 dimensional (b) approach.*

In the early 2000s, hyperbolic space received attention as a potential target space for self-organizing maps (SOMs) with the hyperbolic SOM (HSOM) [WR02]. Although restricted to a grid, the HSOM algorithm is only linear in complexity, requires only input distances, and can be used as a visualization. This algorithm was integrated into a visualization tool [WOWR03] where the SOM lattice was drawn on a Poincaré disk and the data points drawn as 3 dimensional objects coming up out of the plane.

4.3. Torus

Torus-based graph visualizations are relatively recent in comparison to the other non-Euclidean geometries above. Early on, torus embeddings of graphs was primarily of theoretical interest. Since the torus has genus 1, it admits crossing-free drawings of many

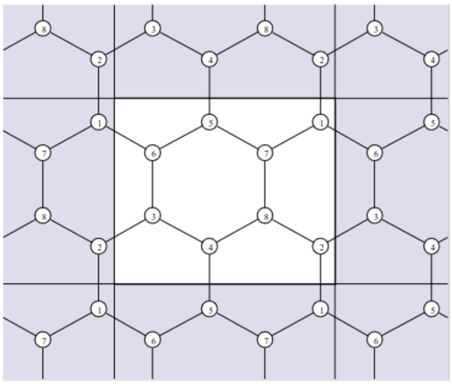


Figure 6: *Torus graph embedding depiction from [KNS01]. This tiling approach is often taken for small graphs on the torus.*

non-planar graphs. An example of a non-planar graph (K_5) drawn without crossings is shown in Fig. 7 [Hea90].

Kocay et al. [KNS01] first describe an algorithm to embed any 2-connected toroidal graph on the torus, where a toroidal graph is one that can be embedded without crossings on the torus. They do this by generalizing Read's algorithm [Rea86] for Euclidean planar graphs, and go on to show that this will always produce a crossing free drawing on the torus. These drawings are depicted as rectangles, with the surrounding area duplicated to show the 'wrapping around' of the torus; see Fig. 6.

This rectangular embedding of the flat torus has been used to create planar drawings of non-planar graphs, by duplicating vertices and edges like the sides of the square in Fig. 6. Biedl [Bie22] shows that any toroidal graph has a visibility representation on a rectangular flat torus, where a visibility representation is an embedding in which vertices are mapped to horizontal line segments and edges are mapped to vertical "lines of sight".

More recently, a series of papers has introduced a practical technique for graph layout and interactive visualization of graphs on the torus. Starting with [CDMB20], Chen et al. identify 3 research questions that they aim to address:

- **RQ1**: develop layout algorithms that take advantage of the extra flexibility of graph layout on the torus;
- **RQ2**: determine how we can best visualize the layout of a node-link diagram on the surface of a torus on a piece of paper or 2D computer monitor; and
- **RQ3**: determine what, if any, perceptual benefits graph layout on a torus has over standard layout on a 2D plane.

This initial paper begins to address these challenges by proposing a stress-based algorithm for torus layout, providing some interaction styles, and a small but thorough user study to evaluate the different interactions and how they compare to traditional drawings. A follow up paper [CDBM21] improved the previous optimization, and took a closer look at cluster based tasks on the torus, performing a user study and found some advantages over traditional drawings (e.g., when identifying clusters).

Most recently, Chen et al. [CNK*23] use eye tracking data to further understand how readers use the toroidal graph drawings to make inferences. They use the tracking data to identify patterns and compare different representations of the torus. They found that duplicating the torus at each of the 8 neighboring grid squares was the most effective, but poorly utilizes space. This leaves a space-efficient full-context torus visualization as future work.

Open Problem: Space-efficient full-context torus visualization

4.4. Other

Focus+context effects can also be achieved with lens effects [TGK*17, TGK*14]. In particular, at first glance the Poincaré disk appears to resemble a fisheye lens, which in some cases is preferable to a flat drawing [Fur86]. However, a lens effect generally modifies a given Euclidean embedding, while hyperbolic or spherical geometries are a global change. Abello et al. combine hierarchical clustering with a fish-eye lens to create a focus+context graph lens [AKY04]. Gansner et al. provide a topological graph lens, based on a hierarchy of coarsened graphs [GKN05]. Wang et al. [WWZ*19] propose a structure-aware fish-eye lens that brings focus to focal area or areas, while maintaining some constraints such as drawing shape and preventing node overlap. A comparison with Du et al.'s spherical technique [DCL*17] and a generic hyperbolic technique is given, indicating that this structure aware lens may outperform non-Euclidean techniques for path-following tasks.

5. Contribution

Our second dimension of analysis is contribution. We discuss papers based on what we believe is their primary contribution. The four broad categories are *techniques, evaluation, theory, and application*. It is common for any visualization research paper to have some mix of a few of these, and so several papers fall into two or more categories.

5.1. Technique

The classical straight-line graph layout problem is to assign 2 dimensional Euclidean coordinates to the vertices of the graph so that the resulting drawing captures well the underlying graph properties (e.g. preserving distances, minimizing crossings, preserving symmetries, etc.). This assignment is known as an *drawing*. Nodes are drawn as circles centered at the desired position, and edges are straight-line segments connecting the corresponding endpoints. There are many different approaches for producing Euclidean graph drawings; the survey by Gibson et al. includes a good selection [GFV13].

The straight-line graph drawing (also called graph layout) problem naturally generalizes to Riemannian spaces, by instead finding a drawing in the desired space with the straight line requirement becoming geodesic curves. We identify four distinct approaches to solving this problem algorithmically.

5.1.1. Projection-Reprojection

The first approach is to have the algorithm take as input a Euclidean drawing of the graph, and project the positions into the

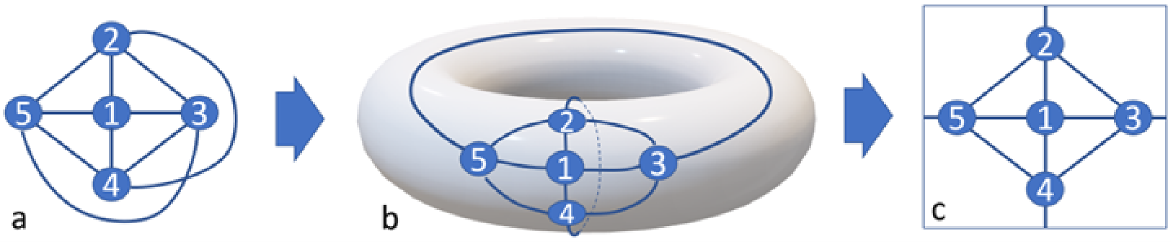


Figure 7: *The canonical non-planar graph K_5 can be drawn without crossings on the torus, as shown in this example from [CDMB20].*

desired space. This is particularly intuitive in spherical space, where geographic map projections from the sphere to the Euclidean plane been studied since classical times. This is one of the approaches presented by Perry et al. [PYGK20], referred to as the “Projection–Reprojection" method. Here, any Euclidean visualization (e.g. node-link diagram) can be drawn on the sphere, by treating the input as a sheet of paper, and ‘folding’ it around the curved sphere surface; see Fig. 8. A similar technique for the sphere is presented in [DCL*17] and another is adapted to hyperbolic space in [MKH22]. Any conceivable node-link based graph visualization idiom can be adapted to a non-Euclidean space using this technique, including map-style drawing [GHK10] or BubbleSets [CPC09], as it only requires a set of coordinates and lines to draw between them. Additionally, this method requires only linear time (given a pre-computed input) and is straightforward to implement. However, this method does not use any unique aspect of the target geometry (such as the ‘wrap-around’ of the sphere, or exponential expansion of hyperbolic space), as shown in [PYGK20]. This method works for any space which admits a projection to Euclidean (since a projection is linear, it has an inverse), but we are unaware of any work which attempts such a method for the torus.

Open Problem: Torus visualization based on the projection-reprojection approach.

5.1.2. Tangent planes

The second technique is to exploit the fact that Riemannian geometries are locally flat, and perform a force-directed style layout algorithm in the tangent spaces about the vertices, explained in detail below. This is the approach in [KW05] and generalizes layout algorithms driven by forces. When computing the forces on a vertex, first compute the tangent plane about that point (a plane which, in 3 dimensions, intersects the surface at exactly one point). Project the other vertices into that plane, and compute forces and update the central vertex position as normal. Finally, project the central vertex back into the target space. Repeatedly apply until the quality of the drawing is sufficient. The tangent plane technique has been reapplied in [DCL*17, MKH22]. While this approach makes use of some unique properties of the geometry, this method is computationally quite expensive and does not achieve as high of quality as the later approaches [MKH22]. While the formulation is for a general Riemannian geometry, we are unaware of any implementation for the torus.

Open Problem: Torus visualization based on the tangent planes approach.

5.1.3. Constrained optimization

The third technique requires us to view our target space as the 2 dimensional surface of a 3 dimensional object. Then, this surface is defined by a set of 3 dimensional Euclidean coordinates that satisfy some property or equation (e.g. a unit sphere is all points with magnitude exactly one). One can perform a force-based graph layout technique in 3 dimensions, requiring that any movement of the vertices still satisfies the properties of the desired space.

This particular technique has been applied only to the sphere, first described by [DLM09] for dimension reduction and adapted to graph layout by [PYGK20]. This is also the approach taken by [SKOM12], who show this is related to information diffusion models, such as independent cascade. Rodighiero [Rod20] describes this constraint as ‘gravity’ attracting the nodes to a spherical surface. The Fruchterman-Reingold algorithm was also adapted in this fashion [GF22]. Although this is easy to describe at a high level, the technical details are quite difficult. One must ensure the step size is sufficiently small, errors will quickly accumulate in the 3 dimensional representation, and the underlying mathematics is quite complicated.

This approach could be extended to hyperbolic space and the torus, but has not been done so. For instance, just like the equation $x^2+y^2+z^2=1$ describes a sphere in 3 dimensions with components x,y,z the equation $x^2+y^2-z^2=1$ describes a hyperbolic surface.

Open Problem: Torus and hyperbolic visualization based on constrained optimization.

5.1.4. Native formulation

Something common to the previous three methods is that they all require the use of the friendly properties of Euclidean space. It is desirable to solve the problem entirely within the target geometry, for fewer computations, simplicity, and in order to fully benefit from the advantages of the particular geometry (e.g., more space in hyperbolic geometry).

The tree drawing algorithms discussed for hyperbolic geometry [MB95, LRP95, SSGR18] are native formulations but they are limited to trees or spanning trees of graphs. Similarly, there are algorithms to generate crossing free drawings of toroidal graphs on the torus [KNS01], but again these are limited to toroidal graphs.

For general graphs, a native formulation typically requires generalizing some Euclidean graph layout approach. The most popular

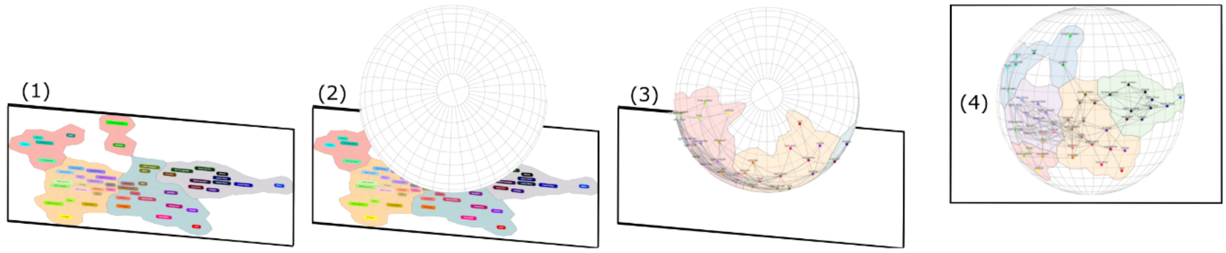


Figure 8: *Example from [PYGK20], illustrating the Projection-Reprojection drawing method for the sphere. By treating the input Euclidean drawing as a sheet of paper, the spherical drawing can be obtained by 'wrapping' the input around the ball.*

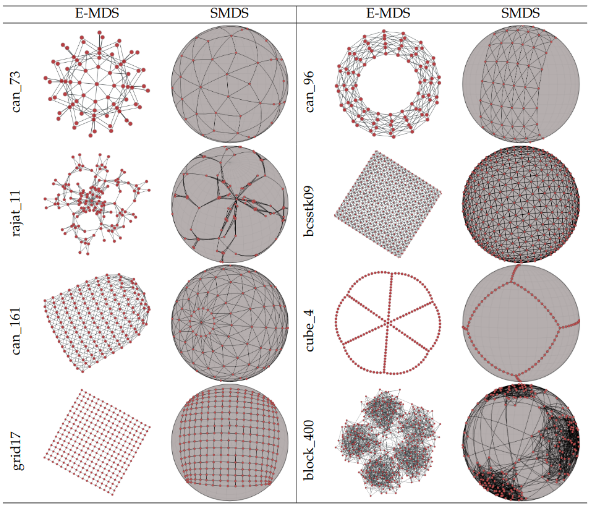


Figure 9: *Several graphs shown drawn by Euclidean MDS and SMDS [MHK23] to show the large difference the choice of drawing space makes.*

choice for this has been stress minimization, also known as Multi-dimensional Scaling (MDS) [Kru64]. Stress-based approaches aim to match the graph theoretic distance between all pairs of vertices to the geodesic distance between the corresponding pairs of nodes in the drawing.

This was first done in the sphere for dimension reduction by [EKK05], then adapted to graph drawing with an improved optimization by [MHK23]. The basic idea is straightforward: replace the L_2 norm computation in Euclidean MDS with the spherical geodesic distance function. Since positions on the sphere are completely defined by a pair of angles, and this function is differentiable, one can find a minimum without the cumulative error found in the constrained optimization approaches. A table taken from [MHK23] is shown in Figure 9.

Classical MDS has been explored in hyperbolic space, by replacing the conversion to similarities with an appropriate hyperbolic scaling function [CE17, SSGR18]. Using a similar idea to the spherical MDS, metric and non-metric MDS have been generalized to hyperbolic space by incorporating hyperbolic geodesic distance into the cost function [WOWR03, Wal04, WR02, ZS21, MKH22].

While [CDMB20] use a stress minimization scheme to compute their toroidal layouts, the torus is missing a closed-form distance function. Chen et al. overcome this by checking which of the straight-line distances is closest to the ideal for each pair of vertices, but a toroidal layout that uses closed form functions is an interesting direction for future work.

Open Problem: Toroidal graph layout that uses a closed-form function.

5.2. Evaluation

Here we focus on papers whose primary contribution is an evaluation of non-Euclidean graph layouts, and also, more generally, typical ways in which such layouts are evaluated.

5.2.1. Quality Measures

The evaluation in most technique and application papers is done by mean of quantitative metrics: metrics that evaluate a layout or drawing with a single number. Time in seconds to produce the drawing is popular, but there are many metrics which measure how "good" a drawing is i.e. how readable or how faithful. These are known as aesthetic and faithfullness metrics respectively, detailed in subsequent subsections.

Such quality metrics were originally defined and studied in the Euclidean setting.

Aesthetic metrics: Properties such as number of crossings, crossing angle, angular resolution, vertex resolution, symmetry, and average edge length are all desirable to optimize for readability, with number of crossings being particularly impactful [Pur02]. These concepts can extend to non-Euclidean geometries, but there is not much work in this area. Eppstein shows that the vertex and angular resolution is bounded in hyperbolic space, meaning one cannot achieve a good resolution in general [Epp21]. In other words for hyperbolic drawings drawn at a realistic scale (cannot be approximated by the Euclidean plane), there exits planar graphs that admit only planar drawings with exponentially small angular resolution. Even restricted classes such as grid graphs only admit drawings with polynomially small angular resolution. The torus admits smaller crossing numbers in theory [KNS01], and this has been shown to be achievable for some real world graphs [CDBM21]. However, much remains to be explored in this area. For example, hyperbolic spaces can have symmetries not possible in Euclidean space , but there is no proposed method to quantitatively measure hyperbolic symmetry.

Open Problem: Formulation of new aesthetic criteria for layouts based on affordances of non-Euclidean geometry.

Faithfulness metrics: Faithfulness metrics instead measure how well the drawing captures the information present in the data [NEH13]. These include metrics such as stress [GKN04], neighborhood preservation [KRM*17], and distortion [KPK*10].

In the graph drawing literature, the normalized stress of a layout is a standard quality measure [GHN13,KRM*17,ZCH*21], defined as:

$$\sum_{i<j} \frac{(||X_i - X_j|| - d_{i,j})^2}{(d_{i,j})^2} \tag{1}$$

where X_i is the embedded position of vertex i, and $d_{i,j}$ is the desired distance between vertex i and vertex j.

This is perfectly acceptable in Euclidean space where a layout is not meaningfully changed when the layout is resized. For non-Euclidean graph layouts there is a possible issue of *dilation or resizing*. Formally, a dilation is a function on a metric space M, $f: M \to M$ that satisfies $d(f(x), f(y)) = rd(x,y)$ for $x, y \in M$, $r > 0 \in \mathbb{R}$ and $d(x,y)$ being the distance between x and y.

In non-Euclidean spaces the size of a layout can have drastic effects since they have an absolute scale [MHK23]. For this reason, *distortion* [SSGR18, MKH22] should be used as a quality metric in place of the normalized stress, defined as

$$\binom{V}{2}^{-1} \sum_{i<j} \frac{|\,||X_i - X_j|| - d_{i,j}|}{d_{i,j}} \tag{2}$$

Distortion is less sensitive to the dilation of the drawing, but note that selecting the correct size is still important to finding a good drawing. Stress has also been generalized to the torus in [CDMB20], by replacing the distance computation $||X_i - X_j||$ with one that finds the minimum distance in all 9 possible wrappings i.e.

$$\sum_{i<j} d_{i,j}^{-2} (\min_{1 \le q \le 9} ||X_i - X_j^q|| - d_{i,j})^2 \tag{3}$$

where q designates which cell to wrap towards if you tile the torus as squares; see Fig. 6 for an example of such a tiling. Distortion can be generalized to the torus, but no work so far makes use of it for toroidal drawings.

Open Problem: Use of distortion as an evaluation metric for graph drawing in non-Euclidean geometries.

Other quality metrics such as neighborhood preservation have yet to be studied in non-Euclidean spaces.

Open Problem: Adapting traditional metrics such as neighborhood preservation to graphs embedded in non-Euclidean geometries.

5.2.2. User studies

While less common, an important form of evaluation for non-Euclidean graph visualization is the user study. A survey of the literature on human-centered experiments in graph visualization shows little work in non-Euclidean spaces [YAD*18]. Several papers incorporate some form of user study as part of the evaluation/justification for a technique or application [CDMB20, SG12] but there are fewer papers dedicated to user evaluation.

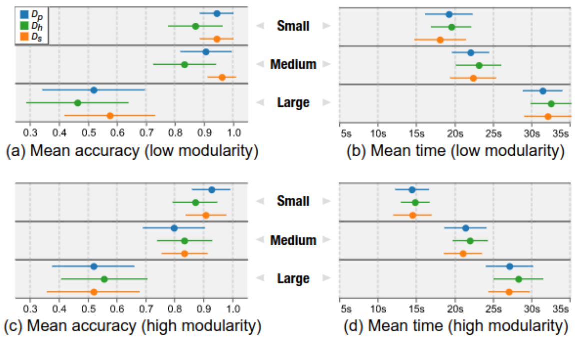


Figure 10: *Plots from [DCL*17], time (right) and error (left) of participant results. The top row are results for low modularity graphs (weak clustering) and the bottom for high modularity graphs (strong clustering). The colors indicate the display geometry: blue for Euclidean, green for hyperbolic, orange for spherical.*

Du et al. [DCL*17] compare Euclidean, spherical, and hyperbolic graph drawings by performing a user study. They first present their iSphere system for large graph visualization, which takes any Euclidean, straight line drawing, and performs an inverse Stereographic (conformal) projection on it to map it to the sphere. They present participants with layouts of stochastic block model random graphs, and ask them questions about some nodes or groups of nodes. The experiment design was within-subjects, each participant saw examples of Euclidean, spherical, and hyperbolic displays, supporting zooming and panning. The layout was driven by MDS for all 3 displays: standard for Euclidean, inverse stereographic projection for sphere, and [KW05] for hyperbolic. All the tasks are about node degrees in the drawings. In particular, they ask for: (1) the neighbor of highest degree for a highlighted node; (2) the common neighbor of highest degree for two highlighted nodes; and (3) the highest degree node along a highlighted path. The sizes of graphs were also controlled: $|V|$ ranged from {128, 512, 2048} and $|E|$ from {1024, 4096, 16384}. The number of clusters generated was 3 for each graph. For each size of graph, they used two different modularities, low and high (a graph statistic which expresses how dense the clusters in a graph are). They conclude with statistical analysis on their collected participant data. The results seemed to indicate that the sphere and Euclidean visualizations had comparable time and accuracy, while hyperbolic performed worse (see Fig. 10).

Meanwhile, for the torus a thorough user study was conducted in [CDBM21]. This study was primarily aimed at asking whether toroidal visualizations were more beneficial for cluster identification. The graph sizes used in the study ranged from $|V| \in (68{,}134)$ with $|E| \in (710,\ 2590)$. Similar to [DCL*17], they divide their graphs into small and large classes, and into low and high modularity. Graphs are randomly generated from stochastic block model type generators.

The authors registered several hypotheses on the open science forum, such as that the torus-based visualization would be beneficial over Euclidean drawings in identifying the number of clusters (in time and error) when the graph modularity is low (i.e. when clusters are not very dense). They performed a within-subjects study so that all participants see all variations; namely both the torus and Euclidean layouts with small/large, low/high modularity, and 2 tasks. The two tasks were: (1) What is the number of clusters; and (2) Do the two highlighted nodes belong to the same cluster?

The statistical analysis of the participant data provides several insights. They found statistically significant results that torus-based layouts increased participant accuracy over Euclidean layout in some cases; we include a figure showing the mean time and accuracy from their study in Fig. 11.

Recently, Chen et al. [CDY*22] compare the typical flat node-link diagram to a torus visualization, along with different types of sphere projections. Finding from this study include that interactive panning improves accuracy for all sphere projections, and that for the torus, equal-earth projection and orthographic projection outperform a flat node-link diagram in cluster identification tasks.

5.3. Proof/theory

Planar graphs are of particular interest in spherical graph visualization. Aleardi et al. [ADF18] develop an algorithm that embeds a planar graph on the sphere without crossings. The algorithm is efficient, but can generate very short edges, resulting in occasionally undesirable layouts with highly varying edge lengths. The authors present several use cases, including as an initialization for a Euclidean layout.

Kryven et al. introduce a measure of visual complexity, *spherical cover number* of a drawing, which is the minimum number of spheres needed to cover all edges of a 3 dimensional graph drawing (sim. circles for 2 dimensions) [KRW19].

Kang and Lin [KL21] show that planar Cayley graphs (graphs that describe a finite group and operations on that group, and is also planar) admit symmetrical spherical drawings. These drawings reside in $\mathbb{S}^2$, are rotationally symmetric, and have uniform edge lengths. They provide a construction for each drawing and enumerate all planar Cayley graphs.

A generalization of the straight line drawing for the Euclidean plane is the geodesic drawing for the sphere, where all edges must be geodesics of the endpoints. This type of drawing has been studied by [BMW20].

Mohar first showed how one can draw general graphs in the hyperbolic plane [Moh99]. Note also that there are theoretical limits on the effectiveness of hyperbolic drawings for general graphs. Some graphs can be embedded trivially with a low, constant embedding error (e.g., as cycles and square lattices) but have non-trivial embedding error in the hyperbolic plane [Epp21, VS16], where embedding error refers to some a measure of distortion of the data. However, other graphs such as trees and hyperbolic tilings can be embedded better in hyperbolic space than in Euclidean space. For example, while Euclidean geometry only admits 3 regular tessellations (triangles, squares, hexagons), the hyperbolic plane admits infinitely many.

Although drawing of arbitrary graphs have theoretical limitations, there are still many graph structures that cannot be well captured in Euclidean space but can be well captured in hyperbolic space. Ouyang et al. [OCCZ12] construct arbitrary hyperbolic tesselations, and provide an algorithm to color and distort edges in aesthetically pleasing and interesting patterns.

It has been shown that some graphs can be embedded with lower error in hyperbolic space than in Euclidean space [BFKL18]. Zhou and Sharpee [ZS21] show that hyperbolic MDS (H-MDS) can be used to detect the underlying geometry of a dataset, when comparing its embedding error to Euclidean metric MDS. They go further to show that the underlying space of genomes is hyperbolic. Krioukov et al's [KPK*10] work indicates that hyperbolic geometry may underlie complex networks and hierarchical networks, such as phylogenetic trees and the internet.

Hyperbolic geometry is of interest in networking and routing, in the form of greedy embeddings. It has been shown that any connected, finite graph admits a greedy embedding in hyperbolic space, which is not generally true in Euclidean geometry [Kle07]. Greedy embeddings of graphs allow for greedy routing, which is particularly useful when a node may not know the global topology, but only its own position and that of its neighbors such as in social networks and the internet [EG11, EG08].

Cabello et al. [CMS15] study toroidal embeddings: given a disconnected graph $G \cup G'$, does there exist a toroidal embedding of the graph with optimal crossings such that no edge of G intersects an edge of G'. Some results are presented, but the problem appears to remain open. Norine [Nor09] shows that any 4-Pfaffian graph (a graph that can be expressed as a linear combination of 4 Pfaffian graph orientation matrices) can be drawn on the torus such that that every perfect matching intersects itself an even number of times.

Open Problem: Given a disconnected graph $G \cup G'$, does there exist a toroidal embedding of the graph with optimal crossings such that no edge of G intersects an edge of G'?

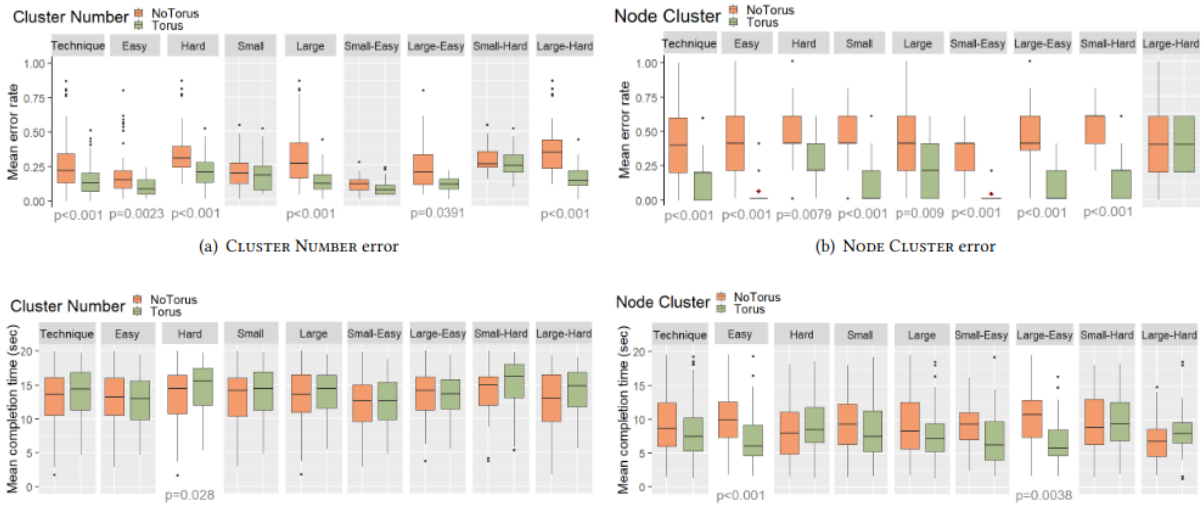
5.4. Application

Applications of network visualization on the sphere are primarily of the form of a map metaphor, or for geographic data.

Shelley et al. [SG12] present a software titled GerbilSphere, which places a user in the center of a sphere with a network drawn on the sphere surface. The target is dynamic, large-scale networks so the tasks are primarily high level movement of groups. The paper includes a small user study to test the effectiveness, reporting that navigation is more efficient with GerbilSphere than traditional network visualization.

Chen provides several examples of toroidal and spherical drawings of real-world graphs in Chapter 2 of [CHE22].

Lambert et al. [LBA10] use a 3D edge bundling technique based on edge routing to support visualization of geographical networks as shown in Fig. 12 where they visualize international air traffic network. The width of the bundles is based on the number of edges within it, and the data is overlayed on the globe since the data has geographic coordinates.



(a) Cluster Number error

(b) Node Cluster error

(c) Cluster Number time

(d) Node Cluster time

Figure 11: *Box plots of results from [CDBM21], showing that Torus based visualization was often more accurate than NoTorus (Euclidean) visualization of graphs for cluster counting task.*

Figure 12: *Using 3D edge routing around the globe to visualize international air interconnections network from [LBA10].*

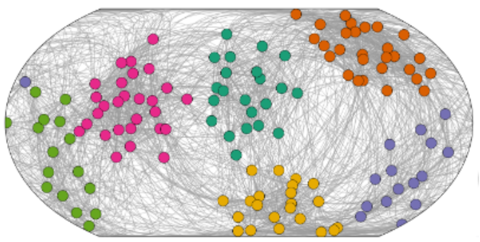


Figure 13: *A cluster graph drawn on the sphere, shown in the equal-earth projection from [CDY*22].*

Efforts to visualize the difficult to conceptualize processes in non-Euclidean geometry has had some interest. Francis and Sullivan [FS04] create a visualization algorithm and rendering to visualize a sphere eversion (turning it inside-out). Further, hyperbolic tessellations have been the subject of study for education and aesthetics [OCCZ12]. An open-source hyperbolic visualization tool, RogueViz [CK17], includes different projections and educational tools, although its restriction to tessellations of the hyperbolic plane makes it less than ideal for general graphs.

Non-Euclidean geometry is of particular interest in virtual reality. This has been explored for the sphere by Kwon et al. [KMLM15, KMLM16], who present a technique for edge routing on the sphere with VR in mind, running the edges outside of the sphere. A user is placed inside the sphere with a head-mounted display, and is free to look around to gain understanding of the network data. In [KDB23], a user is placed in the center of a spherical drawing dubbed an 'egocentric' view. This also allows areas to be 'retracted' and brought closer to the user.

A class of graph known as a 'torus graph' is common in networking and HPC, in fact some years ago high-dimensional torus networks were used in four of the top ten supercomputers and eight of the top ten on the Graph500 list [MIB*14]. These are graphs which describe the topology of the torus in n dimensions. They are created by starting with an n-dimensional rectangular lattice, then connecting the outside vertices in the same dimension; left to right and top to bottom for a two dimensional torus graph. There has been interest in visualizing these types of networks, and we have classified these works as application papers since they are a method of visualizing data from a torus.

TorusVis [CDJM14] is a visualization design study attempting to visualize the topology of torus networks from super computing clusters. They use a circular layout for the graph visualization part of their dashboard, with an ordering algorithm based on Hilbert curves. They use edge bundling to decrease clutter within the circle. This circular layout was later improved and extended into a technique and analytics tool [CZIM18]. Also [MIB*14] look specifically at the IBM Blue Gene machine's torus network using four connected views depicting the network at different levels of detail. The approach is demonstrated by analyzing network traffic for a simulation running on the IBM Blue architecture. Small multiples with links between them are used by [TSW14] to visualize torus networks.

6. Types of Graphs

Our final dimension for analysis is the types of graphs under consideration, as the choice of graphs for visualization in different geometries can affect the visualization methods and goals.

6.1. Trees and Planar graphs

Hyperbolic tree browsers had a large effect on the research landscape of tree and hierarchy visualization. Schneiderman et al. [SDSW12] perform a thorough investigation of early tree visualization techniques, including hyperbolic trees as well as treemaps and cone trees. They show that at the time of writing, the Lamping et al. [LRP95] paper for the hyperbolic browser was one of the most cited tree visualization papers.

Hyperbolic tree visualizations are typically in 2D or 3D. Most 2D approaches employ node-link diagrams drawn in the Poincaré disk [LRP95, HHDK00, BS00]. Meanwhile, most 3D approaches [MB95, Mun97, Mun98, Mun00, Hyu00, ZK16] make use of the Beltrami-Klein projection.

`Treevis.net` provides several examples of hyperbolic and sphere-based tree visualizations [Sch11]. For example, SphereTree [BM12] puts a treemap onto the surface of a sphere, encoding vertices as nested rectangles. The sphere appears hollow, allowing a reader to see the 'backside' of the surface. The hyperbolic wheel [LD12] scheme places a sunburst chart in the Poincaré disk, so that areas of vertices further from the center decrease exponentially.

Planar and non-planar graphs remain planar and non-planar respectively in spherical and hyperbolic geometries. Spherical planar embeddings of graphs are studied by both [BMW20] who aim to eliminate geodesic crossings and [KL21] who exhaustively show a small class of graphs (Planar Cayley graphs) have symmetrical planar spherical embeddings. Planarity can be generalized to higher-genus surfaces. A graph which can be drawn without crossings on the torus is called a toroidal graph. Algorithms to construct crossing-free embeddings of toroidal graphs are known [KNS01], but there are no known efforts to use these algorithms for visualization.

Open Problem: Graph drawing techniques leveraging algorithms to construct crossing-free embeddings of toroidal graphs.

6.2. Complex networks

Krioukov et al. [KPK*10] show that complex networks have an underlying hyperbolic geometry. What this means is that the graph theoretic distance between vertices in a complex network is roughly equivalent to the hyperbolic geodesic distance of a hyperbolic embedding of that graph. One of the ways they show this is by defining a random graph model, a hyperbolic random graph, which uniformly distributes points in the hyperbolic plane and connects points with an edge if they are within some radius. They show this random graph model exhibits complex network properties where many other random graph models (e.g. Euclidean geometric random graphs) do not.

This result lead to a series of papers in hyperbolic graph drawing. The main question is how to generate a hyperbolic graph embedding which realizes the properties above. Bläsius et al. [BFKL18] propose a maximum likelihood estimation formulation for hyperbolic graph embedding, and show computationally that they achieve lower distortion than the force-directed approach of [KW05]. Meanwhile, Sala et al. [SSGR18] develop a hyperbolic classical MDS approach that achieves lower distortion than Euclidean embeddings on example datasets. For metric MDS, Miller et al. [MKH22] apply and stochastic gradient descent optimization to also achieve lower distortion that Euclidean embeddings. Bläsius et al. [BFK21] recently adapted a force-directed approach with two types of forces: popularity force, which operates on the radial component (i.e. its distance from the origin) and similarity force, which operates on the angle of a node. By treating these forces independently, they achieve comparable distortion error to competing methods.

6.3. Highly clustered graphs

A cluster in a graph is a dense subgraph within a larger, typically sparse graph. In other words, a cluster is a collection of vertices in which the number of edges between two vertices in the collection is much greater than the number of edges going out of the collection [Sch07]. Graph density can refer to the ratio of edges to vertices [Law01] or to the ratio defined by the number of edges in the graph and the maximum possible number of edges (in the complete graph) [DD17]. Cluster detection and visualization is of interest, especially in real-world data. The stochastic block model (SBM) is a graph generator that that aims to create clustered graphs [Abb17].

The focus+context effect from non-Euclidean spaces is suitable to visualize the dense regions of highly clustered data, by only showing the focal regions in high detail. Du et al. [DCL*17] use SBM graphs in their study to show that spherical visualization is at least as effective at supporting adjacency tasks as Euclidean. Chen et al. [CDY*22] also use highly clustered graphs and show that both spherical and toroidal drawings are more effective than flat drawings at a cluster identification task. An example spherical drawing is shown in Figure 13.

Most of the algorithms discussed in Section 5.1 adapt distance-based Euclidean techniques such as MDS to Riemannian geometries. However, such distance-based techniques often fail to capture clusters present in the graph [KRM*17]. Algorithms that explicitly aim to optimize cluster separation in a non-Euclidean drawing is an avenue for future work. The popular t-SNE [VdMH08] DR technique is one such algorithm used for high dimensional data, and has been generalized to the hyperbolic plane [GGY22] and the sphere [WW16] but has not been applied to graph drawing.

Open Problem: Graph drawing technique generalizing t-SNE based algorithm to non-Euclidean geometries.

7. Discussion

In this survey we attempt to summarize and analyze work on visualizing graphs in non-Euclidean spaces. We have tagged all papers under consideration and categorized them in a json file with paper titles, authors, DOIs, and other attributes, which we provide as supplemental material. Plots and tables referred to in this section are based on this data.

Geometry	Papers
spherical	18
hyperbolic	19
torus	13

Table 2: *Count of papers in survey that fall into each geometry.*

Looking at the total number of papers over time, grouped by geometry, we see that the three categories have roughly an equal number of papers currently. Most of the papers in hyperbolic geometry are from before 2010, when spherical papers become more popular. Torus papers have been more frequent in recent years and seem to be catching up.

The count of papers by geometry can be found in Table 2. As to be expected, the number of spherical and hyperbolic papers is roughly equal, while there fewer torus papers. We investigate how this trend evolved over time in Fig. 14 (a). From 1995-2009, hyperbolic publications were more numerous with their applications to tree visualization, and received another round of papers in the early 2010s for their relationship to complex networks. Spherical visualization papers became more prevalent in the 2010s as well, perhaps due to the development of map-based network drawings and the release of open source web tools like d3.geo and THREE.js. We see a similar spike in torus papers, when visualization of torus networks for HPC systems became of interest.

Contribution	Papers
technique	16
evaluation	6
proof	13
application	17

Table 3: *Number of Papers based on contribution.*

A count of papers instead by contribution is found in Table 3. Technique, theory, and application papers are roughly equally common. Notably, there are far fewer evaluation papers than one might expect for a field of research originating 30 years ago.

We can observe the intersection of two of the dimensions of our categorization; geometry and contribution. A table that counts the occurrences of these intersections is found in Table 4. A more striking visual is the matrix visualization Fig. 15. We can clearly see where there is potential for future research. Toroidal techniques and evaluations for all geometries have much room to explore (the white spaces in the table).

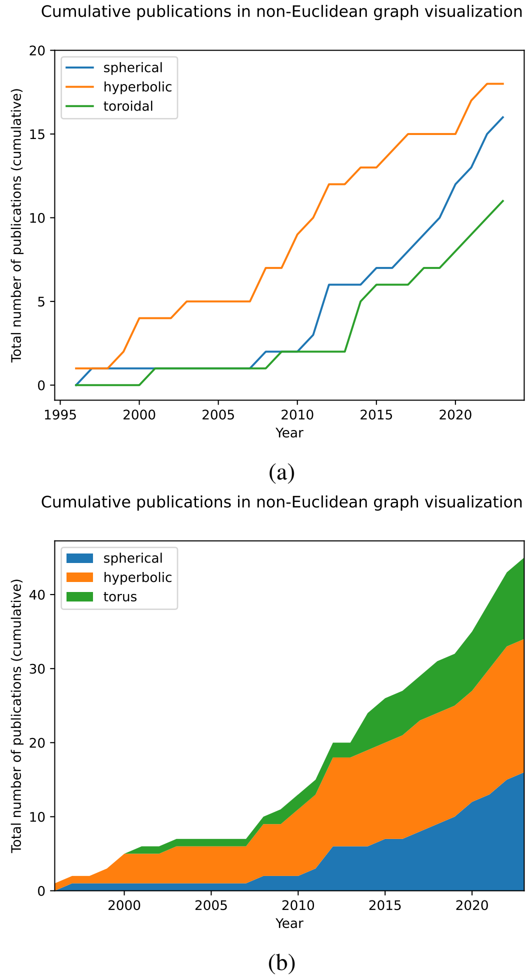


Figure 14: *Number of publications in each geometry over time (cumulative). (a) depicts a simple line chart with total number of publications increasing cumulatively each year. (b) depicts the same data as a stacked area chart, with the height of the chart being the total number of publications of all categories and the colors within the distribution of spherical, hyperbolic, and toroidal geometries.*

	technique	evaluation	proof	application
spherical	6	4	3	5
hyperbolic	5	1	6	6
torus	2	4	4	4

Table 4: *Papers in each geometry based on contribution.*

8. Conclusions and Future work

While there exists a large body of interesting research in this subject area, there is still much to be done. We give an overview of potential future work.

We are unaware of any human-subject studies evaluating graph embedding techniques in spherical and hyperbolic space. For example, is there qualitative difference between the Projection-

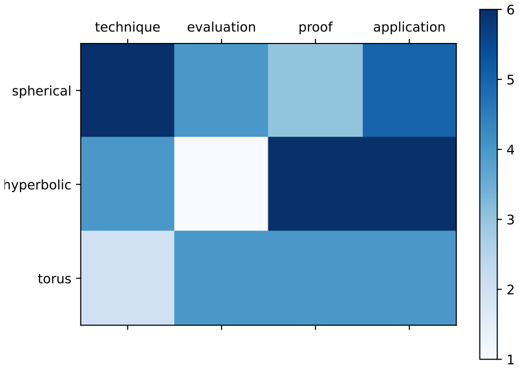


Figure 15: *A matrix based visualization of Table 4. Darker squares indicate more papers published in that category, while lighter squares indicate fewer papers.*

Reprojection methods (which are very fast, but do not take advantage of the underlying geometry) and the more computationally expensive native formulations?

Open Problem: Further computational and human-subjects evaluation of the effect of distortion on non-Euclidean graph drawings.

A large advantage of the sphere as pointed out by [GF22, MHK23] is that there is no defined 'center' or 'periphery'. Any node can be placed in the center of the drawing, something not possible in Euclidean space. Can this be used to prevent anchoring or similar biases?

Open Problem: Evaluate effect of anchoring bias on spherical drawings against Euclidean drawings.

There are many potentially interesting questions in quantitative measures for non-Euclidean geometries. Are there bounds for other aesthetic criteria such as crossing angle like there are for angular resolution in hyperbolic space [Epp21]? Do similar bounds exist for the sphere and torus? Can we characterize the graphs that 'live naturally' (i.e. have low distortion) in hyperbolic/spherical/toroidal space? Do other faithfulness metrics like Neighborhood Preservation need modifying to make sense in non-Euclidean space?

Open Problem: Characterization of graphs which have low distortion in spherical/hyperbolic space.

Open Problem: Adapting traditional metrics such as neighborhood preservation to graphs embedded in non-Euclidean geometries.

Like the spherical and hyperbolic space, the torus has an absolute scale. However, the effect of dilation/resizing of a drawing on the torus has not yet been studied.

Open Problem: Evaluate how the size of input effects graph drawing on torus.

We do not know how people read, understand, and utilize non-Euclidean embeddings. More human-subject studies can provide guidelines for the design of other techniques and applications. While the study by Du et al. [DCL*17] indicates that the sphere may be just as effective as Euclidean space (while hyperbolic is not as effective), this was only shown on a limited set of tasks and using a limited type of graphs. Further studies are needed to verify this result. As we can see in Fig. 15, the evaluation column is light for all three geometries, but hyperbolic is in particular need of verification.

Rodighiero [Rod20] posits that the choice of projection of the sphere effects the reader perception of the network data and argue that a projection that preserves the continuity of the network is desirable. Can we: (1) show that this is true; and (2) select projections more intelligently? This question has been addressed in regards to cluster identification tasks in [CDY*22], finding that equal-earth and orthographic projections indeed outperform flat drawing for this task.

Open Problem: How does the choice of spherical, hyperbolic, or toroidal projection affect task performance?

Many generalizations of graphs, such as multi-layered graphs, dynamic graphs and hypergraphs, have not been explored in Riemannian geometry. Techniques that effectively visualize these types of graphs in new spaces could reveal new insights and seem to be a worthwhile direction for future work.

Open Problem: Multi-layer/dynamic/hypergraph visualization in non-Euclidean spaces.

Although there have been some attempts to utilize non-Euclidean geometry in the virtual reality [KMLM15, KMLM16], this field seems ripe for exploration of immersive visualization. Of course, data can be put on a globe, but the focus+context effect of hyperbolic space is exaggerated in 3D.

Open Problem: Virtual reality techniques and evaluation for non-Euclidean geometries.

We have reviewed over 60 papers in the non-Euclidean and Riemannian graph visualization area, as well as summarized and categorized the literature. This subject has been of interest to the visualization community in the last three decades and we hope that visualization researchers might use this survey to get an overview of what is known and what remains to be explored..

Acknowledgements

This work was partially supported by NSF grant CCF-2212130.

References

[Abb17] ABBE E.: Community detection and stochastic block models: Recent developments. *J. Mach. Learn. Res. 18* (2017), 177:1–177:86. URL: http://jmlr.org/papers/v18/16-480.html. 13

[ADF18] ALEARDI L. C., DENIS G., FUSY É.: Fast spherical drawing of triangulations: An experimental study of graph drawing tools. In *17th International Symposium on Experimental Algorithms, SEA 2018, June 27-29, 2018, L'Aquila, Italy* (2018), D'Angelo G., (Ed.), vol. 103 of *LIPIcs*, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pp. 24:1–24:14. URL: https://doi.org/10.4230/LIPIcs.SEA.2018.24, doi:10.4230/LIPICS.SEA.2018.24. 2, 11

[AKY04] ABELLO J., KOBOUROV S. G., YUSUFOV R.: Visualizing large graphs with compound-fisheye views and treemaps. In *International Symposium on Graph Drawing* (2004), Springer, pp. 431–441. 7

[APN07] ANDREWS K., PUTZ W., NUSSBAUMER A.: The hierarchical visualisation system (HVS). In *11th International Conference on Information Visualisation* (2007), IEEE Computer Society, pp. 257–262. 6

[BBDW17] BECK F., BURCH M., DIEHL S., WEISKOPF D.: A taxonomy and survey of dynamic graph visualization. *Comput. Graph. Forum 36*, 1 (2017), 133–159. URL: https://doi.org/10.1111/cgf.12791, doi:10.1111/cgf.12791. 1

[Bel] BELMONTE N. G.: Javascript InfoVis Toolkit. https://philogb.github.io/jit/demos.html. accessed: 2021-06-06. 6

[BFK21] BLÄSIUS T., FRIEDRICH T., KATZMANN M.: Force-directed embedding of scale-free networks in the hyperbolic plane. In *19th International Symposium on Experimental Algorithms, SEA 2021, June 7-9, 2021, Nice, France* (2021), Coudert D., Natale E., (Eds.), vol. 190 of *LIPIcs*, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pp. 22:1–22:18. URL: https://doi.org/10.4230/LIPIcs.SEA.2021.22, doi:10.4230/LIPICS.SEA.2021.22. 13

[BFKL18] BLÄSIUS T., FRIEDRICH T., KROHMER A., LAUE S.: Efficient embedding of scale-free graphs in the hyperbolic plane. *IEEE/ACM Trans. Netw. 26*, 2 (2018), 920–933. 11, 13

[Bie22] BIEDL T.: Visibility representations of toroidal and klein-bottle graphs. In *Graph Drawing and Network Visualization - 30th International Symposium, GD 2022, Tokyo, Japan, September 13-16, 2022, Revised Selected Papers* (2022), Angelini P., von Hanxleden R., (Eds.), vol. 13764 of *Lecture Notes in Computer Science*, Springer, pp. 404–417. URL: https://doi.org/10.1007/978-3-031-22203-0_29, doi:10.1007/978-3-031-22203-0_29. 2, 7

[BKP*12] BÖRNER K., KLAVANS R., PATEK M., ZOSS A. M., BIBERSTINE J. R., LIGHT R. P., LARIVIÈRE V., BOYACK K. W.: Design and update of a classification system: The UCSD map of science. *PLOS ONE 7*, 7 (07 2012), 1–10. URL: https://doi.org/10.1371/journal.pone.0039464, doi:10.1371/journal.pone.0039464. 1, 2, 5

[BM12] BRATH R., MACMURCHY P.: Sphere-based information visualization: Challenges and benefits. In *16th International Conference on Information Visualisation, IV 2012, Montpellier, France, July 11-13, 2012* (2012), Banissi E., Bertschi S., Forsell C., Johansson J., Kenderdine S., Marchese F. T., Sarfraz M., Stuart L. J., Ursyn A., Wyeld T. G., Azzag H., Lebbah M., Venturini G., (Eds.), IEEE Computer Society, pp. 1–6. URL: https://doi.org/10.1109/IV.2012.11, doi:10.1109/IV.2012.11. 1, 5, 13

[BMW20] BONAMY M., MOHAR B., WESOLEK A.: Limiting crossing numbers for geodesic drawings on the sphere. In *Graph Drawing and Network Visualization - 28th International Symposium, GD 2020, Vancouver, BC, Canada, September 16-18, 2020, Revised Selected Papers* (2020), Auber D., Valtr P., (Eds.), vol. 12590 of *Lecture Notes in Computer Science*, Springer, pp. 341–355. URL: https://doi.org/10.1007/978-3-030-68766-3_27, doi:10.1007/978-3-030-68766-3_27. 2, 11, 13

[Bou] BOU B.: Treebolic2 Webpage. http://treebolic.sourceforge.net/treebolic2/en/index.html. accessed: 2021-06-06. 6

[BP06] BRANDES U., PICH C.: Eigensolver methods for progressive multidimensional scaling of large data. In *Graph Drawing, 14th International Symposium, GD 2006, Revised Papers* (2006), Kaufmann M., Wagner D., (Eds.), vol. 4372 of *Lecture Notes in Computer Science*, Springer, pp. 42–53. URL: https://doi.org/10.1007/978-3-540-70904-6_6, doi:10.1007/978-3-540-70904-6_6. 3

[BS00] BINGHAM J., SUDARSANAM S.: Visualizing large hierarchical clusters in hyperbolic space. *Bioinform. 16*, 7 (2000), 660–661. 6, 13

[BW02] BAUMGARTNER J., WAUGH T. A.: Roget2000: a 2D hyperbolic tree visualization of Roget's thesaurus. In *Visualization and Data Analysis* (2002), vol. 4665 of *SPIE Proceedings*, pp. 339–346. 6

[Bö12] BÖRNER K.:, 2012. URL: http://scimaps.org/home.html. 5

[Cay59] CAYLEY A.: Iv. a sixth memoir upon quantics. *Philosophical Transactions of the Royal Society of London*, 149 (1859), 61–90. 4

[CDBM21] CHEN K., DWYER T., BACH B., MARRIOTT K.: It's a wrap: Toroidal wrapping of network visualisations supports cluster understanding tasks. In *CHI '21: CHI Conference on Human Factors in Computing Systems* (2021), ACM. URL: https://doi.org/10.1145/3411764.3445439. 1, 2, 7, 9, 10, 12

[CDJM14] CHENG S., DE P., JIANG S. H., MUELLER K.: Torusvis$^{\mathrm{nd}}$: unraveling high-dimensional torus networks for network traffic visualizations. In *Proceedings of the First Workshop on Visual Performance Analysis, VPA '14, New Orleans, Louisiana, USA, November 16-21, 2014* (2014), Bremer P., Mohr B., Schulz M., Pascucci V., (Eds.), IEEE, pp. 9–16. URL: https://doi.org/10.1109/VPA.2014.7, doi:10.1109/VPA.2014.7. 12

[CDMB20] CHEN K., DWYER T., MARRIOTT K., BACH B.: Doughnets: Visualising networks using torus wrapping. In *CHI '20: CHI Conference on Human Factors in Computing Systems* (2020), ACM. 1, 2, 7, 8, 9, 10

[CDY*22] CHEN K., DWYER T., YANG Y., BACH B., MARRIOTT K.: Gan'sda wrap: Geographic and network structured data on surfaces that wrap around. In *CHI '22: CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 29 April 2022 - 5 May 2022* (2022), Barbosa S. D. J., Lampe C., Appert C., Shamma D. A., Drucker S. M., Williamson J. R., Yatani K., (Eds.), ACM, pp. 135:1–135:16. URL: https://doi.org/10.1145/3491102.3501928, doi:10.1145/3491102.3501928. 11, 12, 13, 15

[CE17] CLOUGH J. R., EVANS T. S.: Embedding graphs in lorentzian spacetime. *PloS one 12*, 11 (2017), e0187301. 9

[CHE22] CHEN K.-T.: It's a Wrap! Visualisations that Wrap Around Cylindrical, Toroidal, or Spherical Topologies. URL: https://bridges.monash.edu/articles/thesis/It_s_a_Wrap_Visualisations_that_Wrap_Around_Cylindrical_Toroidal_or_Spherical_Topologies/20723092, doi:10.26180/20723092.v1. 1, 11

[CK17] CELINSKA D., KOPCZYNSKI E.: Programming languages in GitHub: A visualization in hyperbolic plane. In *Proceedings of the Eleventh International Conference on Web and Social Media* (2017), AAAI Press, pp. 727–728. 12

[CMS15] CABELLO S., MOHAR B., ŠÁMAL R.: Drawing a disconnected graph on the torus (extended abstract). *Electron. Notes Discret. Math. 49* (2015), 779–786. URL: https://doi.org/10.1016/j.endm.2015.06.105, doi:10.1016/J.ENDM.2015.06.105. 11

[CNK*23] CHEN K., NGO Q. Q., KURZHALS K., MARRIOTT K., DWYER T., SEDLMAIR M., WEISKOPF D.: Reading strategies for graph visualizations that wrap around in torus topology. In *Proceedings of the 2023 Symposium on Eye Tracking Research and Applications, ETRA 2023, Tübingen, Germany, 30 May 2023 - 2 June 2023* (2023), Kasneci E., Shic F., Khamis M., (Eds.), ACM, pp. 67:1–67:7. URL: https://doi.org/10.1145/3588015.3589841, doi:10.1145/3588015.3589841. 7

[CPC09] COLLINS C., PENN G., CARPENDALE S.: BubbleSets: Revealing set relations with isocontours over existing visualizations. *IEEE Trans. Vis. Comput. Graph. 15*, 6 (2009), 1009–1016. 8

[CZIM18] CHENG S., ZHONG W., ISAACS K. E., MUELLER K.: Visualizing the topology and data traffic of multi-dimensional torus interconnect networks. *IEEE Access 6* (2018), 57191–57204. URL: https://doi.org/10.1109/ACCESS.2018.2872344, doi:10.1109/ACCESS.2018.2872344. 1, 2, 13

[DCL*17] DU F., CAO N., LIN Y., XU P., TONG H.: iSphere: Focus+context sphere visualization for interactive large graph exploration. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing* (2017), ACM, pp. 2916–2927. 1, 2, 4, 7, 8, 10, 13, 15

[DD17] DIESTEL R., DIESTEL R.: Extremal graph theory. *Graph theory* (2017), 173–207. 13

[DLM09] DE LEEUW J., MAIR P.: Multidimensional scaling using majorization: SMACOF in R. *Journal of statistical software 31* (2009), 1–30. 5, 8

[EG08] EPPSTEIN D., GOODRICH M. T.: Succinct greedy graph drawing in the hyperbolic plane. *CoRR abs/0806.0341* (2008). URL: http://arxiv.org/abs/0806.0341, arXiv:0806.0341. 11

[EG11] EPPSTEIN D., GOODRICH M. T.: Succinct greedy geometric routing using hyperbolic geometry. *IEEE Trans. Computers 60*, 11 (2011), 1571–1580. 11

[EKK05] ELBAZ A. E., KELLER Y., KIMMEL R.: Texture mapping via spherical multi-dimensional scaling. In *Scale Space and PDE Methods in Computer Vision* (2005), vol. 3459 of *Lecture Notes in Computer Science*, Springer, pp. 443–455. 5, 9

[Epp21] EPPSTEIN D.: Limitations on realistic hyperbolic graph drawing. In *Graph Drawing and Network Visualization - 29th International Symposium, GD 2021, Tübingen, Germany, September 14-17, 2021, Revised Selected Papers* (2021), Purchase H. C., Rutter I., (Eds.), vol. 12868 of *Lecture Notes in Computer Science*, Springer, pp. 343–357. 2, 9, 11, 15

[Eul41] EULER L.: Solutio problematis ad geometriam situs pertinentis. *Commentarii academiae scientiarum Petropolitanae* (1741), 128–140. 3

[EV89] ESCHER M. C., VERMEULEN J. W.: Escher on escher: Exploring the infinite. 5

[FS04] FRANCIS G. K., SULLIVAN J. M.: Visualizing a sphere eversion. *IEEE Trans. Vis. Comput. Graph. 10*, 5 (2004), 509–515. URL: https://doi.org/10.1109/TVCG.2004.33, doi:10.1109/TVCG.2004.33. 12

[Fur86] FURNAS G. W.: Generalized fisheye views. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 1886, Boston, Massachusetts, USA, April 13-17, 1986* (1986), Mantei M. M., Orbeton P., (Eds.), ACM, pp. 16–23. URL: https://doi.org/10.1145/22627.22342, doi:10.1145/22627.22342. 7

[GF22] GUHA A., FEIBUSH E.: Layout and display of network graphs on a sphere. In *Advances in Computer Graphics - 39th Computer Graphics International Conference, CGI 2022, Virtual Event, September 12-16, 2022, Proceedings* (2022), Magnenat-Thalmann N., Zhang J., Kim J., Papagiannakis G., Sheng B., Thalmann D., Gavrilova M. L., (Eds.), vol. 13443 of *Lecture Notes in Computer Science*, Springer, pp. 67–78. URL: https://doi.org/10.1007/978-3-031-23473-6_6, doi:10.1007/978-3-031-23473-6_6. 8, 15

[GFV13] GIBSON H., FAITH J., VICKERS P.: A survey of two-dimensional graph layout techniques for information visualisation. *Information visualization 12*, 3-4 (2013), 324–357. 7

[GGY22] GUO Y., GUO H., YU S. X.: CO-SNE: dimensionality reduction and visualization for hyperbolic data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022* (2022), IEEE, pp. 11–20. URL: https://doi.org/10.1109/CVPR52688.2022.00011, doi:10.1109/CVPR52688.2022.00011. 1, 6, 13

[GHK10] GANSNER E. R., HU Y., KOBOUROV S.: GMap: Visualizing graphs and clusters as maps. In *IEEE Pacific Visualization Symposium PacificVis* (2010), IEEE Computer Society, pp. 201–208. 8

[GHN13] GANSNER E. R., HU Y., NORTH S. C.: A maxent-stress model for graph layout. *IEEE Trans. Vis. Comput. Graph. 19*, 6 (2013), 927–940. URL: https://doi.org/10.1109/TVCG.2012.299, doi:10.1109/TVCG.2012.299. 10

[GKN04] GANSNER E. R., KOREN Y., NORTH S.: Graph drawing by stress majorization. In *International Symposium on Graph Drawing* (2004), Springer, pp. 239–250. 3, 10

[GKN05] GANSNER E. R., KOREN Y., NORTH S. C.: Topological fisheye views for visualizing large graphs. *IEEE Transactions on Visualization and Computer Graphics 11*, 4 (2005), 457–468. 7

[Gla] GLATZOFER M.: d3-hypertree Github page. https://github.com/glouwa/d3-hypertree. accessed: 2021-06-06. 6

[Gla18] GLATZHOFER M.: *Hyperbolic Browsing*. Master's thesis, Institute of Interactive Systems and Data Science (ISDS),Graz University of Technology, Austria, 2018. 6

[Gla19] GLATZHOFER M.: Hyperbolic tree of life. https://hyperbolic-tree-of-life.github.io/, 2019. 1, 5, 6

[GSF97] GROSS M. H., SPRENGER T. C., FINGER J.: Visualizing information on a sphere. In *Proceedings of VIZ'97: Visualization Conference, Information Visualization Symposium and Parallel Rendering Symposium* (1997), IEEE, pp. 11–16. 1, 5

[GT01] GROSS J. L., TUCKER T. W.: *Topological graph theory*. Courier Corporation, 2001. 3

[Gü07] GÜNTHER I.:, 2007. URL: https://world-processor.com. 5

[Hat] HATCH D.:. URL: http://www.plunk.org/~hatch/HyperbolicApplet/. 2

[Hea90] HEAWOOD P. J.: Map color theorems. *Quant. J. Math. 24* (1890), 332–338. 4, 7

[HHDK00] HAO M. C., HSU M., DAYAL U., KRUG A.: Web-based visualization of large hierarchical graphs using invisible links in a hyperbolic space. In *Advances in Visual Information Management, proceedings of the Fifth Working Conference on Visual Database Systems (VDB5), Fukuoka, Japan, May 10-12, 2000* (2000), Arisawa H., Catarci T., (Eds.), vol. 168 of *IFIP Conference Proceedings*, Kluwer, pp. 83–94. 6, 13

[Hyu00] HYUN Y.:. https://www.caida.org/catalog/software/walrus/#H2540, 2000. Accessed: 2021-06-06. 6, 13

[KDB23] KOBINA P., DUVAL T., BRISSON L.: The spherical retractable bubble space: An egocentric graph visualization throughout a retractable visualization space. *Inf. 14*, 10 (2023), 531. URL: https://doi.org/10.3390/info14100531, doi:10.3390/INFO14100531. 12

[KL21] KANG M., LIN W.: Equilateral spherical drawings of planar cayley graphs. *J. Graph Algorithms Appl. 25*, 1 (2021), 97–119. URL: https://doi.org/10.7155/jgaa.00551, doi:10.7155/JGAA.00551. 11, 13

[Kle07] KLEINBERG R.: Geographic routing using hyperbolic space. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications* (2007), IEEE, pp. 1902–1909. 11

[KMLM15] KWON O., MUELDER C., LEE K., MA K.: Spherical layout and rendering methods for immersive graph visualization. In *2015 IEEE Pacific Visualization Symposium, PacificVis 2015, Hangzhou, China, April 14-17, 2015* (2015), Liu S., Scheuermann G., Takahashi S., (Eds.), IEEE Computer Society, pp. 63–67. URL: https://doi.org/10.1109/PACIFICVIS.2015.7156357, doi:10.1109/PACIFICVIS.2015.7156357. 1, 12, 15

[KMLM16] KWON O., MUELDER C., LEE K., MA K.: A study of layout, rendering, and interaction methods for immersive graph visualization. *IEEE Trans. Vis. Comput. Graph. 22*, 7 (2016), 1802–1815. 5, 12, 15

[KNS01] KOCAY W. L., NEILSON D., SZYPOWSKI R.: Drawing graphs on the torus. *Ars Comb. 59* (2001). 7, 8, 9, 13

[KPK*10] KRIOUKOV D., PAPADOPOULOS F., KITSAK M., VAHDAT A., BOGUÑÁ M.: Hyperbolic geometry of complex networks. *Physical Review E 82*, 3 (2010), 036106. 1, 10, 11, 13

[KRM*17] KRUIGER J. F., RAUBER P. E., MARTINS R. M., KERREN A., KOBOUROV S. G., TELEA A. C.: Graph layouts by t-sne. *Comput. Graph. Forum 36*, 3 (2017), 283–294. URL: https://doi.org/10.1111/cgf.13187, doi:10.1111/cgf.13187. 3, 10, 13

[Kru64] KRUSKAL J. B.: Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika 29*, 1 (1964), 1–27. 9

[KRW19] KRYVEN M., RAVSKY A., WOLFF A.: Drawing graphs on few circles and few spheres. *J. Graph Algorithms Appl. 23*, 2 (2019), 371–391. URL: https://doi.org/10.7155/jgaa.00495, doi:10.7155/JGAA.00495. 11

[KSP23] KALE B., SUN M., PAPKA M. E.: The state of the art in visualizing dynamic multivariate networks. In *Computer Graphics Forum* (2023), vol. 42, Wiley Online Library, pp. 471–490. 1

[KW05] KOBOUROV S., WAMPLER K.: Non-Euclidean spring embedders. *IEEE Trans. Vis. Comput. Graph. 11*, 6 (2005), 757–767. 5, 6, 8, 10, 13

[Law01] LAWLER E. L.: *Combinatorial optimization: networks and matroids*. Courier Corporation, 2001. 13

[LBA10] LAMBERT A., BOURQUI R., AUBER D.: 3D edge bundling for geographical data visualization. In *2010 14th International Conference Information Visualisation* (2010), pp. 329–335. doi:10.1109/IV.2010.53. 11, 12

[LC10] LECUN Y., CORTES C.: MNIST handwritten digit database. URL: http://yann.lecun.com/exdb/mnist/ [cited 2016-01-14 14:24:11]. 1

[LD12] LAM H.-C., DINOV I. D.: Hyperbolic wheel: A novel hyperbolic space graph viewer for hierarchical information content. *International Scholarly Research Notices 2012* (2012). 1, 13

[LRP95] LAMPING J., RAO R., PIROLLI P.: A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In *Human Factors in Computing Systems, CHI '95 Conference Proceedings* (1995), ACM/Addison-Wesley, pp. 401–408. 1, 6, 8, 13

[MB95] MUNZNER T., BURCHARD P.: Visualizing the structure of the world wide web in 3D hyperbolic space. In *Procedings of the 1995 Symposium on Virtual Reality Modeling Language* (1995), ACM, pp. 33–38. 1, 6, 8, 13

[MGM*19] MCGEE F., GHONIEM M., MELANÇON G., OTJACQUES B., PINAUD B.: The state of the art in multilayer network visualization. *Comput. Graph. Forum 38*, 6 (2019), 125–149. URL: https://doi.org/10.1111/cgf.13610, doi:10.1111/cgf.13610. 1

[MHK23] MILLER J., HUROYAN V., KOBOUROV S.: Spherical graph drawing by multi-dimensional scaling. In *Graph Drawing and Network Visualization: 30th International Symposium, GD 2022, Tokyo, Japan, September 13–16, 2022, Revised Selected Papers* (2023), Springer, pp. 77–92. 2, 5, 9, 10, 15

[MIB*14] MCCARTHY C. M., ISAACS K. E., BHATELE A., BREMER P., HAMANN B.: Visualizing the five-dimensional torus network of the IBM blue gene/q. In *Proceedings of the First Workshop on Visual Performance Analysis, VPA '14, New Orleans, Louisiana, USA, November 16-21, 2014* (2014), Bremer P., Mohr B., Schulz M., Pascucci V., (Eds.), IEEE, pp. 24–27. URL: https://doi.org/10.1109/VPA.2014.10, doi:10.1109/VPA.2014.10. 1, 12, 13

[MKH22] MILLER J., KOBOUROV S., HUROYAN V.: Browser based hyperbolic visualization of graphs. In *IEEE Pacific Visualization Symposium PacificVis* (2022), IEEE Computer Society. 2, 6, 8, 9, 10, 13

[Moh99] MOHAR B.: Drawing graphs in the hyperbolic plane. In *Graph Drawing, 7th International Symposium, GD'99, Stirín Castle, Czech Republic, September 1999, Proceedings* (1999), Kratochvíl J., (Ed.), vol. 1731 of *Lecture Notes in Computer Science*, Springer, pp. 127–136. URL: https://doi.org/10.1007/3-540-46648-7_13, doi:10.1007/3-540-46648-7_13. 11

[Mun97] MUNZNER T.: H3: laying out large directed graphs in 3D hyperbolic space. In *IEEE Symposium on Information Visualization* (1997), IEEE Computer Society, pp. 2–10. 1, 6, 13

[Mun98] MUNZNER T.: Exploring large graphs in 3D hyperbolic space. *IEEE Computer Graphics and Applications 18*, 4 (1998), 18–23. 6, 13

[Mun00] MUNZNER T.: *Interactive visualization of large graphs and networks*. PhD thesis, Stanford University, 2000. 1, 6, 13

[NEH13] NGUYEN Q. H., EADES P., HONG S.: On the faithfulness of graph visualizations. In *IEEE Pacific Visualization Symposium, PacificVis 2013, February 27 2013-March 1, 2013, Sydney, NSW, Australia* (2013), Carpendale S., Chen W., Hong S., (Eds.), IEEE Computer Society, pp. 209–216. URL: https://doi.org/10.1109/PacificVis.2013.6596147, doi:10.1109/PacificVis.2013.6596147. 10

[NMSL19] NOBRE C., MEYER M. D., STREIT M., LEX A.: The state of the art in visualizing multivariate networks. *Comput. Graph. Forum 38*, 3 (2019), 807–832. URL: https://doi.org/10.1111/cgf.13728, doi:10.1111/cgf.13728. 1

[Nor09] NORINE S.: Drawing 4-pfaffian graphs on the torus. *Comb. 29*, 1 (2009), 109–119. URL: https://doi.org/10.1007/s00493-009-2354-0, doi:10.1007/S00493-009-2354-0. 11

[OB08] OSINSKA V., BALA P.: Classification visualization across mapping on a sphere. In *New Trends in Multimedia and Network Information Systems*, Zgrzywa A., Choros K., Sieminski A., (Eds.), vol. 181 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2008, pp. 95–106. URL: https://doi.org/10.3233/978-1-58603-904-2-95, doi:10.3233/978-1-58603-904-2-95. 5

[OCCZ12] OUYANG P., CHENG D., CAO Y., ZHAN X.: The visualization of hyperbolic patterns from invariant mapping method. *Comput. Graph. 36*, 2 (2012), 92–100. URL: https://doi.org/10.1016/j.cag.2011.12.005, doi:10.1016/J.CAG.2011.12.005. 11, 12

[Pur02] PURCHASE H. C.: Metrics for graph drawing aesthetics. *J. Vis. Lang. Comput. 13*, 5 (2002), 501–516. URL: https://doi.org/10.1006/jvlc.2002.0232, doi:10.1006/jvlc.2002.0232. 9

[PVM*21] PENG W., VARANKA T., MOSTAFA A., SHI H., ZHAO G.: Hyperbolic deep neural networks: A survey. *IEEE Transactions on pattern analysis and machine intelligence 44*, 12 (2021), 10023–10044. 1, 6

[PYGK20] PERRY S., YIN M. S., GRAY K., KOBOUROV S.: Drawing graphs on the sphere. In *AVI '20: International Conference on Advanced Visual Interfaces* (2020), ACM, pp. 17:1–17:9. 5, 6, 8, 9

[Rea86] READ R. C.: *A new method for drawing a planar graph given the cyclic order of the edges at each vertex*. Faculty of Mathematics, University of Waterloo, 1986. 7

[Rod20] RODIGHIERO D.: Drawing network visualizations on a continuous, spherical surface. In *24th International Conference on Information Visualisation, IV 2020, Melbourne, Australia, September 7-11, 2020* (2020), IEEE, pp. 573–580. URL: https://doi.org/10.1109/IV51561.2020.00097, doi:10.1109/IV51561.2020.00097. 8, 15

[Ros12] ROSENFELD B. A.: *A history of non-Euclidean geometry: Evolution of the concept of a geometric space*, vol. 12. Springer Science & Business Media, 2012. 4

[Sch07] SCHAEFFER S. E.: Graph clustering. *Computer science review 1*, 1 (2007), 27–64. 13

[Sch11] SCHULZ H.-J.: Treevis.net: A tree visualization reference. *IEEE Computer Graphics and Applications 31*, 6 (2011), 11–15. 13

[Sch12] SCHAEFER M.: The graph crossing number and its variants: A survey. *The electronic journal of combinatorics* (2012), DS21–Apr. 4

[SDSW12] SHNEIDERMAN B., DUNNE C., SHARMA P., WANG P.: Innovation trajectories for information visualizations: Comparing treemaps, cone trees, and hyperbolic trees. *Information Visualization 11*, 2 (2012), 87–105. 13

[SG12] SHELLEY D. S., GUNES M. H.: Gerbilsphere: Inner sphere network visualization. *Comput. Networks 56*, 3 (2012), 1016–1028. URL: https://doi.org/10.1016/j.comnet.2011.10.023, doi:10.1016/J.COMNET.2011.10.023. 10, 11

[SK03] SANGOLE A. P., KNOPF G. K.: Visualization of randomly ordered numeric data sets using spherical self-organizing feature maps. *Comput. Graph. 27*, 6 (2003), 963–976. URL: https://doi.org/10.1016/j.cag.2003.08.012, doi:10.1016/J.CAG.2003.08.012. 5

[SKOM12] SAITO K., KIMURA M., OHARA K., MOTODA H.: Graph embedding on spheres and its application to visualization of information diffusion data. In *Proceedings of the 21st World Wide Web Conference, WWW 2012, Lyon, France, April 16-20, 2012 (Companion Volume)* (2012), Mille A., Gandon F., Misselis J., Rabinovich M., Staab S., (Eds.), ACM, pp. 1137–1144. URL: https://doi.org/10.1145/2187980.2188253, doi:10.1145/2187980.2188253. 8

[Sny87] SNYDER J. P.: *Map projections: A working manual*. U.S. Government Printing Office, 1987. 3

[SSGR18] SALA F., SA C. D., GU A., RÉ C.: Representation tradeoffs for hyperbolic embeddings. In *Proceedings of the 35th International Conference on Machine Learning* (2018), vol. 80 of *Proceedings of Machine Learning Research*, PMLR, pp. 4457–4466. 1, 8, 9, 10, 13

[SYPB21] SCHÖTTLER S., YANG Y., PFISTER H., BACH B.: Visualizing and interacting with geospatial networks: A survey and design space. *Comput. Graph. Forum 40*, 6 (2021), 5–33. URL: https://doi.org/10.1111/cgf.14198, doi:10.1111/cgf.14198. 1

[TGK*14] TOMINSKI C., GLADISCH S., KISTER U., DACHSELT R., SCHUMANN H.: A Survey on Interactive Lenses in Visualization. In *EuroVis - STARs* (2014), The Eurographics Association. 7

[TGK*17] TOMINSKI C., GLADISCH S., KISTER U., DACHSELT R., SCHUMANN H.: Interactive lenses for visualization: An extended survey. In *Computer Graphics Forum* (2017), vol. 36(6), Wiley Online Library, pp. 173–200. 7

[TKY11] TOGAWA S., KANENISHI K., YANO Y.: Peer-to-peer file sharing communication detection using spherical SOM visualization for network management. In *Human Interface and the Management of Information. Interacting with Information - Symposium on Human Interface 2011, Held as Part of HCI International 2011, Orlando, FL, USA, July 9-14, 2011, Proceedings, Part I* (2011), Smith M. J., Salvendy G., (Eds.), vol. 6771 of *Lecture Notes in Computer Science*, Springer, pp. 259–267. URL: https://doi.org/10.1007/978-3-642-21793-7_30, doi:10.1007/978-3-642-21793-7_30. 5

[TSW14] THEISEN L., SHAH A., WOLF F.: Down to earth: how to visualize traffic on high-dimensional torus networks. In *Proceedings of the First Workshop on Visual Performance Analysis, VPA '14, New Orleans, Louisiana, USA, November 16-21, 2014* (2014), Bremer P., Mohr B., Schulz M., Pascucci V., (Eds.), IEEE, pp. 17–23. URL: https://doi.org/10.1109/VPA.2014.6, doi:10.1109/VPA.2014.6. 13

[VBW17] VEHLOW C., BECK F., WEISKOPF D.: Visualizing group structures in graphs: A survey. *Comput. Graph. Forum 36*, 6 (2017), 201–225. URL: https://doi.org/10.1111/cgf.12872, doi:10.1111/cgf.12872. 1

[VdMH08] VAN DER MAATEN L., HINTON G.: Visualizing data using t-sne. *Journal of machine learning research 9*, 11 (2008). 13

[VLKS*11] VON LANDESBERGER T., KUIJPER A., SCHRECK T., KOHLHAMMER J., VAN WIJK J. J., FEKETE J.-D., FELLNER D. W.: Visual analysis of large graphs: state-of-the-art and future research challenges. In *Computer graphics forum* (2011), vol. 30, Wiley Online Library, pp. 1719–1749. 1

[VS16] VERBEEK K., SURI S.: Metric embedding, hyperbolic space, and social networks. *Comput. Geom. 59* (2016), 1–12. 11

[Wal04] WALTER J. A.: H-MDS: a new approach for interactive visualization with multidimensional scaling in the hyperbolic space. *Inf. Syst. 29*, 4 (2004), 273–292. 6, 9

[WOWR03] WALTER J., ONTRUP J., WESSLING D., RITTER H.: Interactive visualization and navigation in large data collections using the hyperbolic space. In *Third IEEE International Conference on Data Mining* (2003), IEEE, pp. 355–362. 6, 9

[WR02] WALTER J. A., RITTER H. J.: On interactive visualization of high-dimensional data using the hyperbolic plane. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2002), ACM, pp. 123–132. 6, 9

[WT06] WU Y., TAKATSUKA M.: Visualizing multivariate network on the surface of a sphere. In *Proceedings of the 2006 Asia-Pacific Symposium on Information Visualisation-Volume 60* (2006), pp. 77–83. 1, 5

[WT17] WANG C., TAO J.: Graphs in scientific visualization: A survey. *Comput. Graph. Forum 36*, 1 (2017), 263–287. URL: https://doi.org/10.1111/cgf.12800, doi:10.1111/cgf.12800. 1

[WW16] WANG M., WANG D.: Vmf-sne: embedding for spherical data. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2016), IEEE, pp. 2344–2348. 13

[WWZ*19] WANG Y., WANG Y., ZHANG H., SUN Y., FU C., SEDLMAIR M., CHEN B., DEUSSEN O.: Structure-aware fisheye views for efficient large graph exploration. *IEEE Trans. Vis. Comput. Graph. 25*, 1 (2019), 566–575. URL: https://doi.org/10.1109/TVCG.2018.2864911. 7

[YAD*18] YOGHOURDJIAN V., ARCHAMBAULT D., DIEHL S., DWYER T., KLEIN K., PURCHASE H. C., WU H.-Y.: Exploring the limits of complexity: A survey of empirical studies on graph visualisation. *Visual Informatics 2*, 4 (2018), 264–282. 10

[YJD*18] YANG Y., JENNY B., DWYER T., MARRIOTT K., CHEN H., CORDEIL M.: Maps and globes in virtual reality. *Comput. Graph. Forum 37*, 3 (2018), 427–438. 5

[ZCH*21] ZHU M., CHEN W., HU Y., HOU Y., LIU L., ZHANG K.: Drgraph: An efficient graph layout algorithm for large-scale graphs by dimensionality reduction. *IEEE Trans. Vis. Comput. Graph. 27*, 2 (2021), 1666–1676. URL: https://doi.org/10.1109/TVCG.2020.3030447, doi:10.1109/TVCG.2020.3030447. 10

[ZK16] ZHANG S., KELLEHER A.: H3py Github Page. https://github.com/buzzfeed/pyh3, 2016. Accessed: 2021-06-06. 6, 13

[ZS21] ZHOU Y., SHARPEE T. O.: Hyperbolic geometry of gene expression. *Iscience 24*, 3 (2021), 102225. 2, 9, 11