Report from Dagstuhl Seminar 13151

Drawing Graphs and Maps with Curves

Edited by
Stephen Kobourov[1], Martin Nöllenburg[2], and Monique Teillaud[3]

1 **University of Arizona – Tucson, US,** kobourov@cs.arizona.edu
2 **KIT – Karlsruhe Institute of Technology, DE,** noellenburg@kit.edu
3 **INRIA Sophia Antipolis – Méditerranée, FR,** Monique.Teillaud@inria.fr

Abstract

This report documents the program and the outcomes of Dagstuhl Seminar 13151 "Drawing Graphs and Maps with Curves". The seminar brought together 34 researchers from different areas such as graph drawing, information visualization, computational geometry, and cartography. During the seminar we started with seven overview talks on the use of curves in the different communities represented in the seminar. Abstracts of these talks are collected in this report. Six working groups formed around open research problems related to the seminar topic and we report about their findings. Finally, the seminar was accompanied by the art exhibition *Bending Reality: Where Arc and Science Meet* with 40 exhibits contributed by the seminar participants.

Seminar 07.–12. April, 2013 – www.dagstuhl.de/13151
1998 ACM Subject Classification I.3.5 Computational Geometry and Object Modeling, G.2.2 Graph Theory, F.2.2 Nonnumerical Algorithms and Problems
Keywords and phrases graph drawing, information visualization, computational cartography, computational geometry
Digital Object Identifier 10.4230/DagRep.3.4.34
Edited in cooperation with Benjamin Niedermann

1 Executive Summary

Stephen Kobourov
Martin Nöllenburg
Monique Teillaud

Graphs and networks, maps and schematic map representations are frequently used in many fields of science, humanities and the arts. The need for effective visualization and aesthetically pleasing design is attested by the numerous conferences and symposia on related topics, and a history that is several centuries old. From Mercator's maps dating to the 1500's, to interactive services such as Google Earth, geography and cartography have generated and solved many theoretical and practical problems in displaying spatial data effectively and efficiently. From Euler's visualization of the bridges of Königsberg in the 1700's, to Facebook's social networks, graph drawing has also proven a fertile area for theoretical and practical work. More recent is the notion of highly schematized maps and graphs, with the classic examples of statistical value-by-area cartograms by Raisz and Henry Beck's London Tube map, both dating back to the 1930's.

A key challenge in graph and cartographic visualization is designing cognitively useful spatial mappings of the underlying data that allow people to intuitively understand the

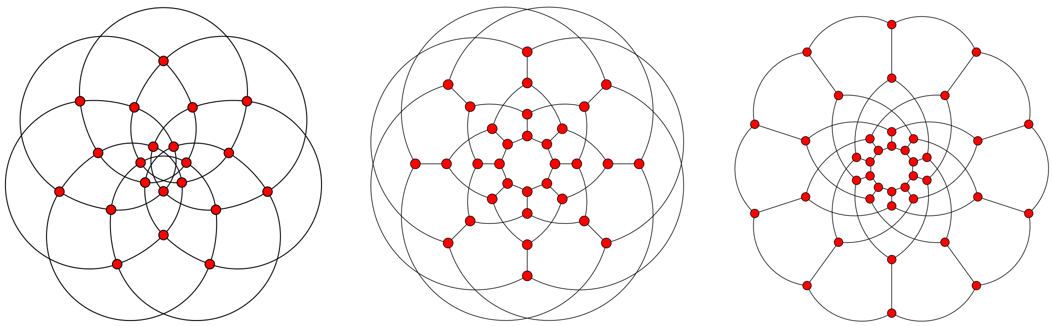


Figure 1 Lombardi graph drawings [1]: Brinkman graph, Dyck graph, and F_{40} (dodecahedron double cover).

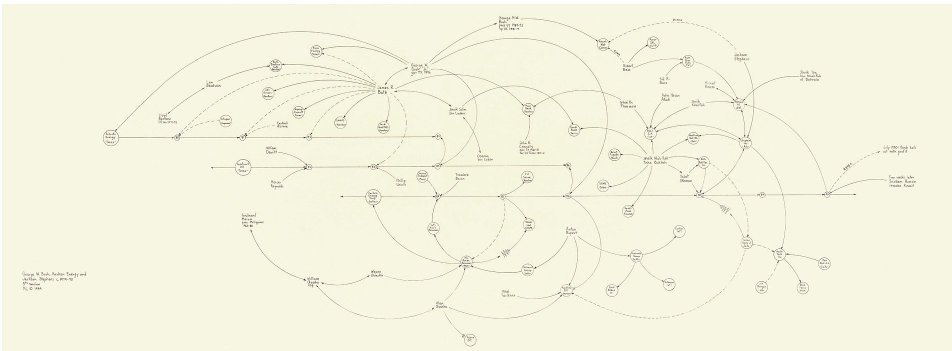


Figure 2 One of Mark Lombardi's pieces: *George W. Bush, Harken Energy, and Jackson Stevens ca. 1979–90*, 1999. Graphite on paper, 20 × 44 inches. Courtesy Pierogi Gallery and Donald Lombardi. Photo credit: John Berens.

displayed information. Such work draws on the intellectual history of several traditions, including information visualization, human-computer interaction, psychology, cognitive science, graphic design, cartography, and art. The synthesis of relevant ideas from these fields with new techniques can lead to new and better visualizations to help us keep pace with the torrents of data confronting us.

Although a great deal is known, both in theory and in practice, about drawing graphs and maps with straight-line segments, there are few corresponding results about circular-arc drawings in particular, and curve drawings in general. The use of circular arcs in place of straight-line segments opens a new chapter in drawing graphs and maps from both theoretical and practical points of view. Specifically, we are interested in the interplay between practical requirements of drawing with curves, arising in cartography and GIS, and theoretical results in computational geometry and graph drawing. Such work is motivated by perception research which indicates that representing paths with smooth geodesic trajectories aids in comprehension, as well as by the aesthetic appeal of smooth curves; see Fig. 1 and Fig. 2.

Aims of the Seminar

The main goal of this seminar was to bring together researchers with interests in drawing graphs and maps with curves coming from information visualization, psychology, cognitive science, human-computer interaction, graph drawing, computational geometry, cartography, and GIS. It follows in a tradition of several previous similarly structured Dagstuhl seminars on graph drawing and map visualization. From April 7th to April 12th a group of 34 junior and senior researchers from eight different countries gathered in Dagstuhl. Being a *small* seminar with a target participation of 30 persons, the seminar was fully booked, which shows that this seemingly narrow topic still raises a lot of interest in the different communities. We all came together to discuss open research questions and engage in new collaborations around visualizations that replace straight lines with circular arcs and curves. This topic opens a great deal of theoretical and practical possibilities and with this in mind, the specific aims of the Dagstuhl seminar were:

- To learn about the state of the art of the use of curves in the different research areas. We invited a small number of survey lectures to define a common ground for interdisciplinary work.
- To organize an exhibition of art and visual designs on the common theme of curves contributed by participants and artists, and use this to stimulate discussion.
- To identify specific theoretical and practical open problems that need to be solved in order to make it possible to draw graphs and maps with circular arcs and curves.
- To form smaller working groups around some of the identified problems and to initiate a collaborative research process for finding answers and solutions to these problems.
- To report about the progress made in the working groups in a plenary session for getting feedback and further input from members of the other groups.
- To continue the joint research efforts beyond the seminar week and eventually publish those results.

Achievements of the Seminar

The achievements in the seminar were numerous and varied. The subsequent chapters of this report summarize the more important ones.

1. On Monday and Tuesday, we enjoyed seven survey lectures; see Section 3 for the abstracts. David Eppstein opened with a broad overview of the use of curves in visualization of graphs and networks. Günter Rote talked about algorithms for approximating polygonal curves by simpler curves and sequences of biarcs. Sylvain Lazard illustrated connections with algebra and geometry when dealing with curves. Jo Wood surveyed the use of curves in cartography and information visualization. Helen Purchase discussed perception theories and empirical studies on the use of curves in visualization, and Maxwell Roberts discussed the question whether curvilinear metro maps have cognitive benefits over traditional straight-line schematic maps. Finally, Monique Teillaud and Michael Hemmer overviewed the history of the open source project CGAL, the Computational Geometry Algorithms Library, and then discussed specific CGAL packages that are relevant for drawing circular arcs and smooth algebraic curves. Beyond the survey and review talks, we also heard a presentation by Wouter Meulemans about the use of curved schematization of geometric shapes, where the results were obtained via a user study of the participants in the seminar.
2. We also had two short impromptu presentations and software demos. In particular, Günter Rote presented an ipelet to transform polygons into splines in the drawing editor ipe. Jan-Henrik Haunert reported about work in progress and showed a demo on morphing polygonal lines so that edge lengths and angles behave as consistently as possible over time.

3. A number of relevant open problems were formulated early in the seminar and six working groups formed around some of the problems. The groups then worked by themselves, formalizing and solving their specific theoretical and practical challenges. Section 4 contains the working group reports summarizing the problem and sketching the current progress made by the groups. Below is a list of the working group topics.
   a. **Smooth Orthogonal Drawings:** What is the complexity of recognizing whether a given 4-planar graph admits a smooth orthogonal drawing of edge complexity 1?
   b. **Confluent Drawing:** What is the complexity of determining whether a given graph has a so-called *strict* confluent drawing?
   c. **Automated Evaluation of Metro Map Usability:** What are good, objective, quantifiable criteria by which curvilinear metro maps can be evaluated? Can such criteria be used so that linear maps can likewise be compared both with each other and also with curvilinear maps?
   d. **Universal Point Sets for Planar Graph Drawings with Circular Arcs:** What can be said about universal point sets for drawing planar graphs if curves are used instead of straight-line segments?
   e. **Labeling Curves with Curved Labels:** How can points on a smooth curve be labeled automatically using curved labels?
   f. **Graphs with Circular Arc Contact Representation:** Which graphs can be represented by contacts of circular arcs?

   The remaining open problems collected during the seminar are listed in Section 5.
4. We had an excellent exhibition entitled “Bending Reality: Where Arc and Science Meet”; see Section 6. This exhibition is the third one in a series of exhibitions that accompany Dagstuhl seminars where aesthetics and art are naturally part of the scientific topics. It was on display from April 8 to April 21, 2013. Moreover, for the first time in Dagstuhl history, this exhibition is made permanently available as a virtual exhibition that can be accessed at **http://www.dagstuhl.de/ueber-dagstuhl/kunst/13151**.

The last three days of the seminar were dedicated to working group efforts. Several of the groups kept their focus on the original problems as stated in the open problem session, while other groups modified and expanded the problems; see Section 4. On the last day of the seminar we heard progress reports from all groups. The results of two of the groups have recently been accepted to international conferences, and we are expecting further research publications to result directly from the seminar.

Arguably the best, and most-appreciated, feature of the seminar was the opportunity to engage in discussion and interactions with experts in various fields with shared passion about curves. The aforementioned exhibition “Bending Reality” helped make the topics of the seminar more visible and raised new questions. In summary, we regard the seminar as a great success. From the positive feedback that we got it is our impression that the participants enjoyed the unique scientific atmosphere at Schloss Dagstuhl and profited from the scientific program. We are grateful for having had the opportunity to organize this seminar and thank the scientific, administrative, and technical staff at Schloss Dagstuhl. We also thank Benjamin Niedermann for helping us to put this report together.

References

1 Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Martin Nöllenburg. Lombardi drawings of graphs. *J. Graph Algorithms and Applications*, 16(1):85–108, 2012.

2 Table of Contents

3 Overview of Talks

3.1 A Brief History of Curves in Graph Drawing

David Eppstein (University California – Irvine)

> "It is not the right angle that attracts me, nor the straight line, hard and inflexible, created by man. What attracts me is the free and sensual curve—the curve that I find in the mountains of my country, in the sinuous course of its rivers, in the body of the beloved woman."
>
> —Oscar Niemeyer [45]

3.1.1 Early Directions

Hand-generated graph drawings have long used curves, independently of graph drawing research. Examples can be found, for instance, in the works of Listing [38] and Peterson [46]. Mark Lombardi (1951–2000) raised hand-made curved graph drawings to an art form with his diagrams of connections between the actors in global financial, criminal, and political conspiracies, or as he called them, *narrative structures* [31]. Hand-drawn graphs have also been the subject of graph drawing research: Plimmer et al. [48] describe techniques for the automated rearrangement of hand-drawn graphs, taking care to preserve the features that make these graphs look hand-drawn such as the curvature of their edges.

The earliest research on the mathematics and algorithmics of curved graph drawing concerned a drawing style that is now called an *arc diagram*. These are drawings in which the vertices are placed on a line, and the edges are drawn on one or more semicircles, either above or below the line. Drawings with these styles were first used by Saaty [54] and Nicholson [44] as a way to formalize and attempt to solve the problem of minimizing crossing numbers of graphs. Finding the minimum number of crossings of a graph in this style turns out to be NP-hard [41], although if crossing-free diagrams exist they may be found by solving a 2-satisfiability problem once the vertex ordering has been determined [17]. Djidjev and Vrt'o [10] and Cimikowski [8] developed heuristics to reduce the number of crossings in drawings of this type without guaranteeing to find the minimum. For *st*-planar graphs (and also for undirected planar graphs) it is always possible to find drawings in this style in which edges are drawn as smooth curves formed of at most two semicircles (oriented left to right in the *st*-planar case) with at most one crossing of the spine on which the vertices are placed [29, 3].

3.1.2 Uses of Curvature

One use of curvature, in drawings with edges that are curved but that do not have inflection points, is to indicate directionality: if each edge is oriented in clockwise from source to destination, then this orientation will be unambiguously indicated by the curve of the edge without having to use other visual techniques to show it such as arrowheads or line thickness variation [22]. This style is a good fit for arc diagrams [49] but for other layout styles, user studies have shown that it can be confusing to readers [34].

Curved edges may also be used to help edges avoid obstacles that a straight edge would run into. This can already be seen in the work of Peterson [46], who used circular arcs to represent distinct edges connecting the same two vertices in a multigraph. In Sugiyama-style layered graph drawing [58], vertices are assigned to layers and edges are subdivided by dummy vertices so that each connection is between consecutive layers. The dummy vertices (which are used to route edges between the other vertices on each layer) may then be used to guide the placement of spline curves so that the edges may be drawn as smooth curves [28]. A simplified version of this spline calculation by Sander [55] was used by him to win the graph drawing contest at GD 1994 GD. Another obstacle-avoiding drawing style, by Dobkin et al. [11], involves placing the vertices of the drawing first, and then in a second stage of the algorithm routing the edges around the vertices. In their algorithm, each edge is routed first as a shortest obstacle-avoiding polyline, which is then smoothed by a spline curve, and finally the curves are adjusted locally to eliminate any intersections with obstacles caused by the smoothing step. There has also been much related work in motion planning on finding smooth curves for a fixed obstacle-avoiding route; see e.g. Lutterkort and Peters [40].

The two ideas of curvature indicating directionality and curvature to avoid obstacles were combined in PivotGraph [60], a system for displaying graphs whose vertices have other numerical or categorical data associated with them. The vertices are placed into a grid using this additional data as Cartesian coordinates, but this placement would lead to many edges that pass directly through vertices if drawn as straight lines. By using curves (oriented clockwise by directionality) they avoid these bad edge-vertex intersections as well as showing the orientation of the edges graphically. In another visualization system, NodeXL [57], a "polar" layout places vertices on concentric circles; curved edges are used to connect vertices on consecutive circles without passing through the inner circle, cf. [1].

3.1.3 Focus + Context

Sarkar and Brown [56] suggested interactive *fisheye views* of graphs that could be used to simultaneously zoom in on a point of interest and showing its surrounding context. The Poincaré model of hyperbolic geometry (with edges drawn as circular arcs) automatically has this effect [37] and has the additional advantage that there is a natural way to morph from one focus to another, "maintaining the mental map". Mohar [42] proves a version of Fáry's theorem (stating that the existence of non-crossing drawings in which each edge follows a geodesic path) for graphs in the hyperbolic plane or on surfaces of negative curvature.

In later work on highlighting points of interest, Wong et al. [62] use edge curvature to bend edges locally away from a part of a drawing without distorting the vertex placements in the drawing. A related technique called *edge plucking* allows interactive user control of local bending of bundles of edges [61].

3.1.4 Edge Complexity

Much research in graph drawing has focused on drawing styles with angular bends but low *curve complexity* (bends per edge). However, graph drawing researchers have long known that bends can be replaced by smoothed curves [16]. Bekos et al. [3] formalize the *edge complexity* of graphs drawn with piecewise-circular-arc smooth curves, as the maximum number of arcs and straight segments per edge. They observe that edge complexity is always within a constant factor of bend complexity but that in many cases it is actually possible to achieve lower edge complexity than bend complexity.

Goodrich and Wagner [30] achieve low edge complexity in a planar graph drawing. They modify the (straight line) planar drawing algorithm of Fraysseix et al. [25] by surrounding each vertex with a protected region of radius proportional to its degree, and placing equally spaced "ports" on the boundary of this region. Spline curves through the ports have constant edge complexity, near-optimal angular resolution, and do not cross. Similar ideas have also been used by Cheng et al. [6] and Duncan et al. [12] for drawings with piecewise circular edges, again achieving high angular resolution.

3.1.5 Force-directed Graph Drawing (Spring Systems)

Force-directed methods have long been a mainstay of practical graph drawing. These methods use forces (which may be visualized as springs) to attract adjacent pairs of vertices and repel other pairs. Despite being somewhat slow they are very flexible, as they allow the implementer great freedom of choice in modifying the system to add other forces beyond these basic attractive and repulsive ones. Force-directed methods began with Tutte [59], who showed that springs can automatically generate planar straight line drawings of planar graphs. Other important early research in this area (also using straight edges) was done by Eades [15], Kamada and Kawai [36], and Fruchterman and Reingold [26].

The combination of curves and forces was made by Brandes and Wagner [5], as part of a system for drawing graphs that represent train systems. In these graphs, vertices represent train stations and edges connect consecutive stops on the same train route. The vertex placement is fixed by the geographic position of the stations, and in many cases involves sets of nearly-collinear stations spaced out along the same train line. However, the edges representing express trains skip some of these vertices, representing local train stations, and (if drawn as straight) would overlap the positions of these stations. The solution of Brandes and Wagner [5] is to use force-directed methods, with forces on the control points of splines, to bend these edges outwards. In another application to train systems, Fink et al. [23] use force-directed drawing to schematize train system maps, by replacing paths of degree-two vertices by spline curves. Bending outwards can also be used in 3d to separate edges of geographic graphs from the Earth's surface [43].

For arbitrary graphs, Finkel and Tamassia [24] place a new vertex in the middle of each edge of a given graph, apply force-directed layout, and then use the new vertices as spline control points. They report that this gives significant improvements in angular resolution and modest improvements in crossings compared to straight line drawings. Similar ideas were used by Chernobelskiy et al [7] to spread out the edges in drawings with circular-arc edges.

3.1.6 Edge Bundling

Edge bundling is a technique that, as initially developed, was used to simplify drawings of large graphs with a hierarchically clustered vertex structure. This technique groups edges that connect the same two clusters (at some level of the hierarchy) into "bundles" drawn as nearly-parallel curves, making the set of edges both visually distinctive and more compact than if they were all drawn separately. This idea was introduced by Holten [32] based on a physical analogy to electrical wiring bundles; it is also closely related to flow maps for numerical geographic data [47], and since the initial work on this style there have been hundreds of successor papers.

Some of the many variations on bundling that have been considered include:

- Non-hierarchical bundling by modeling edges as springs that attract each other [33].
- A circular vertex layout, with unbundled edges outside the circle chosen to minimize crossings, and with edges grouped into bundles inside the circle using a heuristic that attempts to minimize the total amount of ink used for the drawing [27].

- Edge bundling in Sugiyama-style layered drawing [50].
- Forbidding crossings within the edges of a bundle, and routing the edges of each bundle on parallel tracks resembling metro maps, so that individual edges are easier to follow [4].

For a taxonomy of bundling-related curved edge techniques see Riche et al. [53].

3.1.7 Confluent Drawing

Although visually similar to bundled drawing, and often co-cited with it, confluent drawing [9] is semantically very different. In confluent drawing, one represents a graph using *train tracks* (sets of smooth curves meeting at points called *junctions*) rather than drawing the edges as individual curves. Two vertices are adjacent in the underlying graph if and only if they can be connected by a smooth curve through the curves and junctions of the drawing. Thus, each curve carries a set of edges, similar to bundling, but in an unambiguous way.

The graphs with crossing-free confluent drawings form a strict superset of the planar graphs, and include for example the interval graphs [9] and distance-hereditary graphs [20, 35]. A partially ordered set has an upward-planar confluent *Hasse diagram* if and only if its order dimension is at most two [19].

Confluent edge routing (allowing crossings between pairs of confluent tracks) has been combined with Sugiyama-style layered drawing, by finding complete bipartite subgraphs within the sets of edges that connect consecutive layers and replacing these subgraphs by confluent tracks [21]; this style was used for a winning entry in the 2003 graph drawing contest. Confluence has also been used as a way to achieve axis-parallel edges for high-degree planar graphs [52].

3.1.8 Lombardi Drawing

Inspired by the art of Mark Lombardi, graph drawing researchers have defined *Lombardi drawing* to be a very strict layout style in which edges must be drawn as circular arcs, meeting at equal angles at each vertex. This allows plane trees to be drawn in balloon style (with their subtrees drawn recursively in disks surrounding the root node) with polynomial area, in contrast to straight line drawing styles for which drawing plane trees with equally spaced edges at each vertex may sometimes require exponential area [14].

Lombardi drawing may also be used for regular graphs and symmetric graphs [13], planar graphs with maximum degree three [18], and some other special cases [39]. However, not every graph has a Lombardi drawing, causing researchers in this area to resort to force-based approximations [7] or multi-arc relaxations [12]. User studies on the aesthetics and usability of force-based Lombardi-like drawings have had mixed results [63, 51] perhaps indicating that additional constraints (such as restricting edges to arcs that cover less than half of their circles) are necessary to make this style more effective.

A related result is that every 3-connected 4-regular planar graph may be represented as the graph of an arrangement of circles [2]; however, the arcs of this representation will not in general be equally spaced around each vertex.

3.1.9 Conclusions

Curves have been used very widely in graph drawing: almost every method of graph drawing that has been studied, has been studied with curves. There are still many remaining technical challenges (for instance on the edge complexity needed for 1-planar drawings, or on the existence of outerplanar Lombardi drawings) but the biggest future challenge for curved drawing is more general, and still has not been completely achieved: designing methods for curved drawing that are general (applicable to every graph and not just specialized graph classes), usable (as measured in user studies), and consistently beautiful.

References

1 Christian Bachmaier, Hedi Buchner, Michael Forster, and Seok-Hee Hong. Crossing minimization in extended level drawings of graphs. *Discrete Appl. Math.*, 158(3):159–179, 2010.
2 Michael A. Bekos and Chrysanthi N. Raftopoulou. Circle-representations of simple 4-regular planar graphs. In *Graph Drawing 2012*, volume 7704 of *LNCS*, pages 138–149. Springer, 2013.
3 Michael A. Bekos, Michael Kaufmann, Stephen G. Kobourov, and Antonios Symvonis. Smooth orthogonal layouts. In *Graph Drawing 2012*, volume 7704 of *LNCS*, pages 150–161. Springer, 2013.
4 Sergey Bereg, Alexander E. Holroyd, Lev Nachmanson, and Sergey Pupyrev. Edge routing with ordered bundles. In *Graph Drawing 2011*, volume 7034 of *LNCS*, pages 136–147. Springer, 2012.
5 Ulrik Brandes and Dorothea Wagner. Using graph layout to visualize train interconnection data. *J. Graph Algorithms and Applications*, 4(3):135–155, 2000.
6 Christine C. Cheng, Christian A. Duncan, Michael T. Goodrich, and Stephen G. Kobourov. Drawing planar graphs with circular arcs. *Discrete Comput. Geom.*, 25(3):405–418, 2001.
7 Roman Chernobelskiy, Kathryn I. Cunningham, Michael T. Goodrich, Stephen G. Kobourov, and Lowell Trott. Force-directed Lombardi-style graph drawing. In *Graph Drawing 2011*, volume 7034 of *LNCS*, pages 320–331. Springer, 2012.
8 Robert Cimikowski. Algorithms for the fixed linear crossing number problem. *Discrete Appl. Math.*, 122(1-3):93–115, 2002.
9 Matthew T. Dickerson, David Eppstein, Michael T. Goodrich, and Jeremy Yu Meng. Confluent drawings: visualizing non-planar diagrams in a planar way. In *Graph Drawing 2003*, volume 2912 of *LNCS*, pages 1–12. Springer, 2004.
10 Hristo Djidjev and Imrich Vrt'o. An improved lower bound for crossing numbers. In *Graph Drawing 2001*, volume 2265 of *LNCS*, pages 96–101. Springer, 2002.
11 David P. Dobkin, Emden R. Gansner, Eleftherios Koutsofios, and Stephen C. North. Implementing a general-purpose edge router. In *Graph Drawing 1997*, volume 1353 of *LNCS*, pages 262–271. Springer, 1997.
12 Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Maarten Löffler. Planar and poly-arc Lombardi drawings. In *Graph Drawing 2011*, volume 7034 of *LNCS*, pages 308–319. Springer, 2012.
13 Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Martin Nöllenburg. Lombardi drawings of graphs. *J. Graph Algorithms and Applications*, 16(1):85–108, 2012.
14 Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Martin Nöllenburg. Drawing trees with perfect angular resolution and polynomial area. *Discrete Comput. Geom.*, 49(2):183–194, 2013.
15 Peter Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42(11):149–160, 1984.
16 Peter Eades and Roberto Tamassia. Algorithms For Drawing Graphs: An Annotated Bibliography. Technical Report CS-89-09, Computer Science Dept., Brown University, 1989.
17 Alon Efrat, Cesim Erten, and Stephen G. Kobourov. Fixed-location circular arc drawing of planar graphs. *J. Graph Algorithms and Applications*, 11(1):145–164, 2007.
18 David Eppstein. Planar Lombardi drawings for subcubic graphs. In *Graph Drawing 2012*, volume 7704 of *LNCS*, pages 126–137. Springer, 2013.

19 David Eppstein and Joseph A. Simons. Confluent Hasse diagrams. In *Graph Drawing 2011*, volume 7034 of *LNCS*, pages 2–13. Springer, 2012.

20 David Eppstein, Michael T. Goodrich, and Jeremy Yu Meng. Delta-confluent drawings. In *Graph Drawing 2005*, volume 3843 of *LNCS*, pages 165–176. Springer, 2006.

21 David Eppstein, Michael T. Goodrich, and Jeremy Yu Meng. Confluent layered drawings. *Algorithmica*, 47(4):439–452, 2007.

22 Jean-Daniel Fekete, David Wang, Niem Dang, Aleks Aris, and Catherine Plaisant. Overlaying graph links on treemaps. In *IEEE Symp. on Information Visualization, Poster Compendium*, pages 82–83, 2003.

23 Martin Fink, Herman Haverkort, Martin Nöllenburg, Maxwell Roberts, Julian Schuhmann, and Alexander Wolff. Drawing metro maps using Bézier curves. In *Graph Drawing 2012*, volume 7704 of *LNCS*, pages 463–474. Springer, 2013.

24 Benjamin Finkel and Roberto Tamassia. Curvilinear graph drawing using the force-directed method. In *Graph Drawing 2004*, volume 3383 of *LNCS*, pages 448–453. Springer, 2005.

25 Hubert de Fraysseix, János Pach, and Richard Pollack. Small sets supporting Fary embeddings of planar graphs. In *20th ACM Symp. on Theory of Computing*, pages 426–433, 1988.

26 Thomas M. J. Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.

27 Emden R. Gansner and Yehuda Koren. Improved circular layouts. In *Graph Drawing 2006*, volume 4372 of *LNCS*, pages 386–398. Springer, 2007.

28 Emden R. Gansner, Stephen C. North, and Kiem-Phong Vo. DAG—a program that draws directed graphs. *Software: Practice and Experience*, 18(11):1047–1062, 1988.

29 Francesco Giordano, Giuseppe Liotta, Tamara Mchedlidze, and Antonios Symvonis. Computing upward topological book embeddings of upward planar digraphs. In *Proc. Int. Symp. Algorithms and Computation (ISAAC 2007)*, volume 4835 of *LNCS*, pages 172–183. Springer, 2007.

30 Michael T. Goodrich and Christopher G. Wagner. A framework for drawing planar graphs with curves and polylines. *J. Algorithms*, 37(2):399–421, 2000.

31 Robert Hobbs. *Mark Lombardi: Global Networks*. Independent Curators, 2003.

32 Danny Holten. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Trans. Visualization and Computer Graphics*, 12(5):741–748, 2006.

33 Danny Holten and Jarke J. van Wijk. Force-directed edge bundling for graph visualization. *Computer Graphics Forum*, 28(3):983–990, 2009.

34 Danny Holten and Jarke J. van Wijk. A user study on visualizing directed edges in graphs. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems*, pages 2299–2308, 2009.

35 Peter Hui, Michael J. Pelsmajer, Marcus Schaefer, and Daniel Štefankovič. Train tracks and confluent drawings. *Algorithmica*, 47(4):465–479, 2007.

36 Tomihisa Kamada and Satoru Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31(1):7–15, 1989.

37 John Lamping and Ramana Rao. Laying out and visualizing large trees using a hyperbolic space. In *Proc. 7th ACM Symp. on User Interface Software and Technology*, pages 13–14, 1994.

38 Johann Benedikt Listing. *Vorstudien zur Topologie*. Vandenhoeck und Ruprecht, 1848.

39 Maarten Löffler and Martin Nöllenburg. Planar Lombardi drawings of outerpaths. In *Graph Drawing 2012*, volume 7704 of *LNCS*, pages 561–562. Springer, 2013.

40 David Lutterkort and Jörg Peters. Smooth paths in a polygonal channel. In *Proc. 15th ACM Symp. on Computational Geometry*, pages 316–321, 1999.

41 Sumio Masuda, Kazuo Nakajima, Toshinobu Kashiwabara, and Toshio Fujisawa. Crossing minimization in linear embeddings of graphs. *IEEE Trans. Computing*, 39(1):124–127, 1990.

42 Bojan Mohar. Drawing graphs in the hyperbolic plane. In *Graph Drawing 1999*, volume 1731 of *LNCS*, pages 127–136. Springer, 1999.

43 Tamara Munzner, Eric Hoffman, K. Claffy, and Bill Fenner. Visualizing the global topology of the MBone. In *IEEE Symp. on Information Visualization*, pages 85–92, 1996.

44 T. A. J. Nicholson. Permutation procedure for minimising the number of crossings in a network. *Proc. IEE*, 115:21–26, 1968.

45 Oscar Niemeyer. *The Curves of Time: the memoirs of Oscar Niemeyer*. Phaidon, 2000. As quoted at http://en.wikiquote.org/wiki/Oscar_Niemeyer.

46 Julius Peterson. Die Theorie der regulären Graphen. *Acta Mathematica*, 15(1):193–220, 1891.

47 Doantam Phan, Ling Xiao, Ron Yeh, Pat Hanrahan, and Terry Winograd. Flow map layout. In *IEEE Symp. on Information Visualization*, pages 219–224, 2005.

48 Beryl Plimmer, Helen C. Purchase, Hong Yul Yang, Laura Laycock, and James Milburn. Preserving the hand-drawn appearance of graphs. In *Int. Worksh. Visual Languages and Computing*, 2009.

49 A. Johannes Pretorius and Jarke J. van Wijk. Bridging the semantic gap: Visualizing transition graphs with user-defined diagrams. *IEEE Computer Graphics and Applications*, 27(5):58–66, 2007.

50 Sergey Pupyrev, Lev Nachmanson, and Michael Kaufmann. Improving layered graph layouts with edge bundling. In *Graph Drawing 2010*, volume 6502 of *LNCS*, pages 329–340. Springer, 2011.

51 Helen C. Purchase, John Hamer, Martin Nöllenburg, and Stephen G. Kobourov. On the usability of Lombardi graph drawings. In *Graph Drawing 2012*, volume 7704 of *LNCS*, pages 451–462. Springer, 2013.

52 Gianluca Quercini and Massimo Ancona. Confluent drawing algorithms using rectangular dualization. In *Graph Drawing 2010*, volume 6502 of *LNCS*, pages 341–352. Springer, 2011.

53 Nathalie Henry Riche, Tim Dwyer, Bongshin Lee, and Sheelagh Carpendale. Exploring the design space of interactive link curvature in network diagrams. In *Proc. Int. Working Conf. on Advanced Visual Interfaces (AVI '12)*, pages 506–513, 2012.

54 Thomas L. Saaty. The minimum number of intersections in complete graphs. *Proc. National Academy of Sciences*, 52:688–690, 1964.

55 Georg Sander. Graph layout through the VCG tool. In *Graph Drawing 1994*, volume 894 of *LNCS*, pages 194–205. Springer, 1995.

56 Manojit Sarkar and Marc H. Brown. Graphical fisheye views of graphs. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems*, pages 83–91, 1992.

57 Marc A. Smith, Ben Shneiderman, Natasa Milic-Frayling, Eduarda Mendes Rodrigues, Vladimir Barash, Cody Dunne, Tony Capone, Adam Perer, and Eric Gleave. Analyzing (social media) networks with NodeXL. In *Proc. 4th Int. Conf. Communities and Technologies*, pages 255–264, 2009.

58 Kozo Sugiyama, Shôjirô Tagawa, and Mitsuhiko Toda. Methods for visual understanding of hierarchical system structures. *IEEE Trans. Systems, Man, and Cybernetics*, SMC-11 (2):109–125, 1981.

59 William T. Tutte. How to draw a graph. *Proc. London Math. Society*, 13(52):743–768, 1963.

60 Martin Wattenberg. Visual exploration of multivariate graphs. In *Proc. SIGCHI Conf. on Human Factors in Computing Systems*, pages 811–819, 2006.

61 Nelson Wong and Sheelagh Carpendale. Interactive poster: Using edge plucking for interactive graph exploration. In *IEEE Symp. on Information Visualization*, 2005.

62 Nelson Wong, Sheelagh Carpendale, and Saul Greenberg. EdgeLens: an interactive method for managing edge congestion in graphs. In *IEEE Symp. on Information Visualization*, pages 51–58, 2003.

63 Kai Xu. A user study on curved edges in graph visualization. *IEEE Trans. Visualization and Computer Graphics*, 18(12):2449–2556, 2012.

3.2 Algorithms for Curve Approximation

Günter Rote (FU Berlin)

Joint work of Scott Drysdale, Günter Rote, Astrid Sturm
Main reference R.L.S. Drysdale, G. Rote, A. Sturm, "Approximation of an open polygonal curve with a minimum number of circular arcs and biarcs," Computational Geometry, Theory and Applications 41 (2008), 31-47.
URL http://dx.doi.org/10.1016/j.comgeo.2007.10.009

This is a survey of algorithms for approximating a curve by a simpler curve, mostly by a polygon with few edges. In the last part I also mention an algorithm of Drysdale, Rote, and Sturm [1] for smooth (tangent-continuous) approximation by biarcs. I discuss the problem definition, the variations and different objectives, and the pitfalls. I also survey the classical algorithms of Douglas and Peucker; Imai and Iri; and of Chan and Chin.

References

1 R. L. Scot Drysdale, Günter Rote und Astrid Sturm: Approximation of an open polygonal curve with a minimum number of circular arcs and biarcs. *Computational Geometry, Theory and Applications* 41 (2008), 31–47. doi:10.1016/j.comgeo.2007.10.009

3.3 Algebraic Curves in Computational Geometry

Sylvain Lazard (INRIA Grand Est – Nancy, FRA)

Joint work of Yacine Bouzidi, Sylvain Lazard, Marc Pouget, Fabrice Rouillier

I will survey some recent results on the problem of drawing, in a certified way, algebraic curves in the plane given by their implicit equation. I will focus on the problem of computing the topology of such curves and their critical points. The talk will mostly be centered on the presentation of the standard tools and main classes of methods for this problem. I will also present some recent results (joint with Y. Bouzidi, M. Pouget and F. Rouillier) on the problem of computing the critical points of the curve or, more precisely, on the problem of solving bivariate algebraic systems by means of rational univariate representations. I will show that computing such representations can be done in $O(d^8 + d^\tau)$ bit operations modulo polylogarithmic factors, where d is the degree of the input curve and τ is the maximum bitsize of its coefficients. This decreases by a factor d^2 the best known complexity for this problem. I will finally present some experiments and comparisons between state-of-the-art software for computing the topology of plane algebraic curves.

3.4 Curved Lines in Cartography and Information Visualization

Jo Wood (City University – London)

Calling on examples from the history of cartography and information visualization this talk shows where curved lines have been used to carry information in visual depiction. Examples range from terrain analysis, through sketchy rendering, to flows of people, bicycles and money. The role of metaphor in visual depiction in both maps and information visualization is considered and related to the geometric properties of curved lines.

3.5 Some Brief Notes on Perceptual Theories — in Relation to Empirical Studies

Helen C. Purchase (University of Glasgow)

In trying to understand how people understand information visualisations, we can use two sources; empirical data collected through experiments, or theories of perception. Of course, the two are necessarily intertwined: data suggests theory, theory is validated by data. Conducting experiments is costly and the results are not easily generalisable outside the experimental parameters. What we would like to be able to do is predict the effectiveness of a visualisation prior to its use, and existing theories of perception and cognition can help us do this.

This presentation introduces some common theories of perception that can help in pre-experiment assessment of visualisation effectiveness. It takes as a starting point a paper by Peebles (2004) where the effect of a commonly-known visual illusion was investigated in data plots used for comparing the performance of local councils with respect to several issues (e.g. education, leisure, housing). This example demonstrates how existing perceptual theories can be used to predict the effectiveness of a visualisation – and how, in this case, the predictions were confirmed through an empirical study (and as a result, a better visualisation was proposed).

Several perceptual theories are introduced in this presentation, with focus on the Gestalt theories, and visual acutity. These theories are then related to graph drawings, showing, for example, that the existance of non-symmetric edges can overcome the preception of symmetric nodes. The relative visual priority of visual features like colour, shape, texture etc., is demonstrated through the process of 'pop-out'.

Finally, an experiment comparing the effectiveness of graphs with straight line edges vs curved edges is presented as an example of a study explicitly addressing the limits of visual acuity. The results, surprisingly, showed better performance with the straight-line graph, suggesting that visual acuity is not as problematic in graph-reading tasks as might be expected.

3.6 Do We Need Curvilinear Metro Maps?

Maxwell J. Roberts (University of Essex)

The topics of cognitive load and cognitive capacity are related to metro map usage and, in relation to this, the following optimisation criteria are identified: simplicity; coherence; balance; harmony, and topographicity. It is suggested that as long as these criteria are fulfilled, the design ruses do not matter, and that in certain circumstances it may be necessary to design a curvilinear metro map. Three types of such map are identified: freeform Bézier, concentric circles, and Lombardi, each with strengths and weaknesses likely to suit networks with different qualities. Generally, curvilinear maps may be particularly suited to dense, interconnected networks with complex line trajectories, where linear maps have high angular density and poor coherence (e.g., Paris, Tokyo).

3.7 Curves in CGAL

Michael Hemmer (TU Braunschweig) and Monique Teillaud (INRIA Sophia Antipolis – Méditerranée)

CGAL, the Computational Geometry Algorithms Library, followed the evolution of Computational Geometry, progressively offering more and more curved objects and functionality on them.

The talk first quickly overviews the history of the CGAL open source project, the development process and the contents of the library. It recalls how CGAL solves robustness issues, following the so-called exact geometric computing paradigm pioneered by Chee Yap.

Then packages in CGAL that should be relevant within the context of graph drawing with curves are examined. In particular, support for circles, spheres, triangulations and Voronoi diagrams involving circular arcs, planar arrangements of arbitrary algebraic curves and the algebraic kernel are discussed in more detail.

3.8 Curved Schematization – User Study Results

Wouter Meulemans (TU Eindhoven, NL)

Joint work of Arthur van Goethem, Wouter Meulemans, Bettina Speckmann, Jo Wood

We invited the participants of the seminar to also participate in an online user study related to curved schematization. Curved schematization is the task of obtaining a low-complexity representation of a detailed shape using curves. In particular, we use circular arcs. We had three hypotheses: curved schematization is better than straight-line schematization in terms of (1) aesthetics, (2) visual simplicity, and (3) recognizability. To verify these hypotheses in the user study, we had a single algorithm generate schematizations according to four different styles. One style admitted only straight-line edges; the other three styles used circular arcs. The latter differentiated themselves in the degree of "curviness" (measured as central angle) they strive for. The user study itself consisted of three tasks, one for each hypothesis. For each task, the participant was given a random set of questions about either aesthetics, simplicity, or recognizability. The study had 303 participants. We concluded that the three curved styles were preferred over the straight-line style, but there was no clear preference for any particular curved style. Hence we accept hypothesis (1). In terms of simplicity, the straight-line style and the curved style with low curviness (i.e. using arcs that are almost a straight line) were the best. Therefore we reject hypothesis (2). As for recognizability, we observed that the curved styles performed better than the straight-line style in a range of "medium complexity" (approximately 10 arcs). Thus we accept hypothesis (3).

4 Working Groups

4.1 Smooth Orthogonal Drawings

Michael A. Bekos (National TU – Athens)
Martin Gronemann (Universität Köln)
Sergey Pupyrev (University of Arizona – Tucson)

4.1.1 Introduction

Smooth orthogonal drawings were recently introduced in the graph drawing literature as a response to the problem of smoothening orthogonal drawings (of graphs of maximum degree 4), by "replacing" bends with smooth circular arcs. It is expected that such drawings are of improved readability and more aesthetic appeal, since it is widely accepted that bends interrupt the eye movement and require sharp changes of direction. Smooth orthogonal drawings follow the paradigms of traditional orthogonal drawings, in which each vertex is drawn as a point on the plane and each edge is drawn as an alternating sequence of axis-aligned line-segments. However, they also allow for circular arc-segments. We are particularly interested in providing theoretical guarantees about graphs that admit smooth orthogonal drawings of *edge complexity* 1, where the edge complexity of a drawing is given by the maximum number of segments forming any of its edge. Formally, we say that a graph has *smooth complexity* k if it admits a smooth orthogonal drawing of edge complexity at most k.

The work of Bekos, Kaufmann, Kobourov and Symvonis [1] was the first on the problem of creating smooth orthogonal drawings of 4-planar graphs. They presented an infinite class of 4-regular planar graphs generated from the octahedron graph that do not admit smooth orthogonal drawings of edge complexity 1. They also proved that biconnected 4-planar graphs admit drawings of smooth complexity 3. For triconnected 3-planar graphs and Hamiltonian 3-planar graphs, they gave algorithms to produce smooth orthogonal drawings of edge complexity 1. Finally, they showed that there exist graphs whose smooth complexity-1 drawings needs exponential drawing area (assuming a quite restricted drawing model that requires fixed embedding and fixed ports).

4.1.2 Results

As already stated, not all 4-planar graphs admit smooth orthogonal drawings of smooth complexity 1. What if crossings are allowed in the resulting drawing? The following theorem answers this question in a positive way.

▶ **Theorem 1.** *Any graph of maximum degree 4 admits a (not necessary planar) drawing of smooth complexity 1.*

The input graph is augmented such that it is 4-regular. Based on the work of Peterson [2], a cycle cover decomposition is used to create a book-embedding-like drawing. The cycle cover decomposition guarantees that the port constraints are respected. However, the resulting drawing is not necessarily planar.

Bekos et al. [1] have proven that triconnected 3-planar graphs admit drawings of smooth complexity 1. Their approach is based on canonical ordering, hence, the requirement for triconnectivity. We are able to strengthen this result by using the *BC (Block-Cutpoint)* and *SPQR*-tree data structures to decompose the graph into its bi- and triconnected components, respectively.

▶ **Theorem 2.** *Any 3-planar graph admits a (planar) drawing of smooth complexity 1.*

The proof is constructive and provides an algorithm that exploits the special structure of the BC- and SPQR-trees on 3-planar graphs. The approach uses a slightly modified version of the algorithm in Bekos et al. [1] to deal with the triconnected components. Similar to Eppstein [3] who uses these properties to create Lombardi-style drawings of sub cubic graphs, we are able to maintain smooth complexity 1 while reassembling the components into one drawing. However, the approach relies heavily on the degree restriction, hence, does not easily extend to higher degree.

4.1.3 Open Problems & Ongoing Work

While working on the aforementioned problems during the seminar, it seemed tempting to use a book embedding style drawing as a basis for a layout. Especially when considering the Kandinsky model, i.e. one drops the port constraints, the two seem to be related. Besides the results of the seminar, a list of open problems has been compiled.

- What is the complexity of recognizing whether a given 4-planar graph admits a smooth orthogonal drawing of edge complexity 1?
- Is it possible to determine a universal point set so that any 4-planar graph can be embedded on it with a certain smooth complexity (e.g., SC 2 or SC 3)?
- The algorithm that we currently have for drawing any graph of maximum degree 4 with smooth complexity 1 does not take care of the crossings that arise. So, crossing minimization for non planar graphs is of importance.

References

1 Michael A. Bekos, Michael Kaufmann, Stephen G. Kobourov, and Antonios Symvonis. Smooth orthogonal layouts. In *Graph Drawing 2012*, volume 7704 of *LNCS*, pages 150–161. Springer, 2013.

2 Julius Peterson. Die Theorie der regulären Graphen. *Acta Mathematica*, 15(1):193–220, 1891.

3 David Eppstein. Planar Lombardi drawings for subcubic graphs. In *Graph Drawing 2012*, volume 7704 of *LNCS*, pages 126–137. Springer, 2013.

4.2 Confluent Drawing

David Eppstein (University of California, Irvine, US)
Danny Holten (SynerScope BV)
Maarten Löffler (Utrecht University)
Martin Nöllenburg (KIT – Karlsruhe Institute of Technology)
Bettina Speckmann (TU Eindhoven)
Kevin Verbeek (University of California – Santa Barbara)

Confluent drawing [1] is a style of graph drawing in which edges are not drawn explicitly; instead vertex adjacency is indicated by the existence of a smooth path through a system of arcs and junctions that resemble train tracks. These types of drawings allow even very dense graphs, such as complete graphs and complete bipartite graphs, to be drawn in a planar way. We introduce a subclass of confluent drawings, which we call *strict* confluent drawings. Strict confluent drawings are confluent drawings with the additional restrictions that between any pair of vertices there can be *at most one* smooth path, and there cannot be any paths from a vertex to itself. Figure 3 illustrates the forbidden configurations. We believe that these restrictions may make strict drawings easier to read, by reducing the ambiguity caused by the existence of multiple paths between vertices.

Our results are as follows:

- It is NP-complete to determine whether a given graph has a strict confluent drawing. To prove this, we reduce from planar 3-SAT and we use the fact that K_4 has exactly two strict confluent drawings.
- For a given graph, with a given cyclic ordering of its n vertices, there is an $O(n^2)$-time algorithm to find an *outerplanar* strict confluent drawing, if it exists: this is a drawing in a (topological) disk, with the vertices in the given order on the boundary of the disk (see Figure 4). We define a *canonical diagram* which is a unique representation of all possible outerplanar strict confluent drawings, where some of the faces are marked as cliques. We compute a canonical diagram by first computing its junctions, and then computing the arcs between them. From a canonical diagram we can easily derive an outerplanar strict confluent drawing.
- When a graph has an outerplanar strict confluent drawing, an algorithm based on circle packing can construct a layout of the drawing in which every arc is drawn using at most two circular arcs (again, see Figure 4).

The full version of this abstract will appear (under the title Strict Confluent Drawing) in the proceedings of the 21st International Symposium on Graph Drawing 2013. The most pressing problem left open by this research is to recognize the graphs that have outerplanar strict confluent drawings, without imposing a fixed vertex order. Can we recognize these graphs in polynomial time?

Figure 3 (a) A drawing with a duplicate path. (b) A drawing with a self-loop.

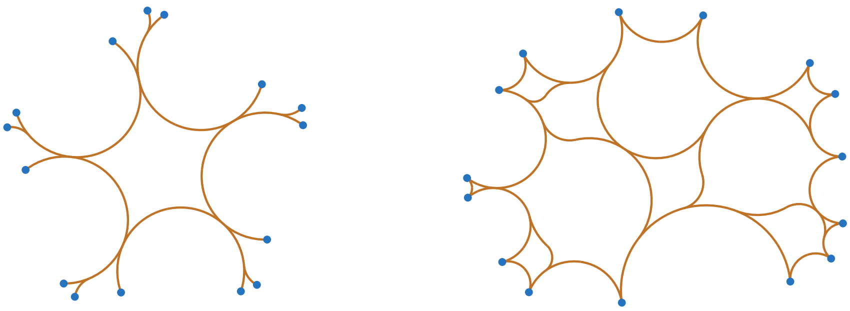


Figure 4 Two outerplanar strict confluent drawings.

References

1 Matthew T. Dickerson, David Eppstein, Michael T. Goodrich, and Jeremy Yu Meng. Confluent drawings: Visualizing non-planar diagrams in a planar way. *Journal of Graph Algorithms and Applications*, 9(1):31–52, 2005.

4.3 Automated Evaluation of Metro Map Usability

Michael Hemmer (TU Braunschweig)
Wouter Meulemans (TU Eindhoven)
Lev Nachmanson (Microsoft Corp. – Redmond)
Helen Purchase (University of Glasgow)
Andreas Reimer (Universität Heidelberg)
Max Roberts (University of Essex)
Günter Rote (FU Berlin)
Kai Xu (Middlesex University)

The problem addressed by this workgroup concerns devising objective quantifiable criteria by which curvilinear metro maps can be evaluated and, subsequently, broadening the applicability of the criteria so that linear maps can likewise be compared both with each other and also with curvilinear maps. In the first instance, the intention was to predict journey planning time differences between maps, as per those revealed in usability studies (for example, comparing freeform Bézier maps with official octolinear designs, for London a curvilinear design is equivalent to the official map, but for Paris a curvilinear design is 50% faster for journey planning than the official map). Ultimately, the development of successful evaluation routines would curtail the need for usability studies.

The methodology adopted was to identify criteria for good design of curvilinear maps, based on exemplars of extreme difference in simplification to line trajectories (a Madrid curvilinear map devised by the first author and one devised by the designers of the current tetralinear official map). From this, the following criteria for good design were identified and attempts made to quantify them independently of each other.

Lower order criteria concern the trajectories of individual lines in isolation of each other. These contribute to the overall simplicity of the design.

- Curve inflections (number and severity).
- Curvature (quantity of change, and severity of change).
- Symmetry of curvature (at Bézier control points).

Higher order criteria concerning relationships between line trajectories. These contribute to the overall coherence of the design.

- Parallel Bézier curves within defined fields.
- Few edge crossings.
- Good angular resolution at edge crossings, especially at line interchanges where the line trajectories are typically part-obscured by the addition of a symbol.

Higher order criteria concerning relationships between line trajectories and stations. These also contribute to the overall coherence of the design.

- Adequate distance between stations and edge crossings (a station too close to an edge crossing is the most frequent source of planning errors in usability studies).
- Continuity of station density (a balanced design should not have abrupt changes in station density in nearby regions of the map).
- Horizontal/vertical alignment of stations.

The first objective was to assemble criteria that would predict differences in basic usability between designs: the mean time necessary to plan a journey (station location time is another variable of interest in evaluating usability). Purely aesthetic criteria were not identified. Aesthetic judgement will be related to the usability criteria outlined above, but other aesthetic criteria (such as a preference for straight lines versus curves) may impact only upon user-acceptance of a design (an important consideration, albeit one that is vulnerable to individual differences in preference). Measures of topographical accuracy were also not implemented. These may affect user acceptance of a design, but their prime influence on usability is their potential to affect journey choices: whether an efficient or inefficient journey is planned. Any such algorithm should penalise a map more for distorted relative positions of nearby stations than distant stations. Finally, the relationship between label placement and usability was not considered. These are an important element of map usability, and future work will address criteria for effective placement.

4.3.1 Specific Criteria Quantifications

Quantified evaluations of lower order criteria for line curvature were fully implemented on an individual line-by-line basis, derived from plots of curvature versus distance. From this, aspects of poor design could easily be identified visually and numerically. The main challenge now is to weight the relative importance of these.

Parallelism and angular resolution were implemented together, the former on the basis of constancy of distance between lines within a defined distance of each other. Angular resolution was incorporated by not subjecting lines to a parallelism analysis if their crossing was greater than a pre-determined angle.

Station density plots were created which enabled hotspots and coldspots to be identified. Designs were penalised on the basis of distance between these. Station placement was evaluated by identifying x-co-ordinates and y-co-ordinates, and for each of these determining the number of aligned stations.

Scores for various criteria were obtained for a variety of curvilinear and linear Madrid maps (high scores are bad for all measures)

1. total inflections
2. overall extent of curvature (variation of curvature over the whole line)
3. symmetry of curvature (sum of the squares of the vertical jumps in curvature–ie: extreme changes in curvature–over the whole line)
4. "lack of parallelism" penalty (not yet normalised for number of lines)
5. number of edge crossings (not yet normalised for complexity = stations x lines)
6. spacing discontinuity of stations
7. vertical/horizontal mis-alignment of station symbols

Simplified curvilinear map: 1) 19; 2) 1.23; 3) 0.006; 4) 5.76; 5) 42; 6) 0.21; 7) 0.19
Tetralinear (rectilinear) map: 1) NYC; 2) NYC; 3) NYC; 4) NYC; 5) 46; 6) 0.26; 7) 0.11
Hexalinear map: 1) NYC; 2) NYC; 3) NYC; 4) NYC; 5) 47; 6) 0.19; 7) 0.16
Complex curvilinear map: 1) 80; 2) 4.02; 3) NYC; 4) 6.94; 5) 42; 6) 0.28; 7) 0.21
Concentric circles map: 1) NYC; 2) NYC; 3) NYC; 4) NYC; 5) 43; 6) 0.18; 7) 0.18
NYC = not yet computed

4.3.2 Future work

Having identified methods of quantification, the next step is to extend them so that relative predictions can be made for linear versus curvilinear maps, and to combine and weight them so that they make predictions that match usability study data. The latter are, unfortunately, relatively course: typically all that can be identified is a relative difference in journey planning time between pairs of maps. This can be mitigated by comparing as may different maps as possible using different design techniques from a variety of cities. Once available data has been modelled, this will then enable future usability studies to be identified so that predictions can be tested and refined. In addition to this, the availability of these criteria will enable particular defects with maps to be identified, and possibly resolved automatically via a simulated annealing process.

4.4 Universal Point Sets for Planar Graph Drawings with Circular Arcs

Patrizio Angelini (Università di Roma III)
David Eppstein (University California – Irvine)
Fabrizio Frati (The University of Sydney)
Michael Kaufmann (Universität Tübingen)
Sylvain Lazard (INRIA Grand Est – Nancy)
Tamara Mchedlidze (KIT – Karlsruhe Institute of Technology)
Monique Teillaud (INRIA Sophia Antipolis – Méditerranée)
Alexander Wolff (Universität Würzburg)

It is well known that every planar graph has a drawing on the plane where vertices are mapped to points of an $O(n) \times O(n)$ integer grid, and edges to straight-line segments connecting the corresponding points. To generalize this result, Mohar (according to Pach [1]) posed the following question: What is the smallest size $f(n)$ (as a function of n) of a point set S such that every n-vertex planar graph G admits a planar straight-line drawing in which the vertices of G are mapped to points in S? Such a point set S is called *universal point set*.

Despite more than twenty years of research efforts, the best known lower bound for the value of $f(n)$ is linear in n, while the best known upper bound is quadratic in n. The question is listed as problem #45 in the Open Problems Project [1].

Given the topic of our seminar, we are interested in understanding whether drawing edges as circular arcs rather than straight-line segments helps to answer the corresponding question. Our starting point is a result of Everett et al. [2] saying that, for any natural number n, there is a universal set of size n if edges are drawn as one-bend polygonal chains.

It turns out that we can get the same result for drawings with circular arcs. More specifically, we prove the existence of a set S of n points on the parabolic arc $\mathcal{P} = \{(x, y)\colon x \geq 0, y = -x^2\}$ such that every n-vertex planar graph G can be drawn such that its vertices are mapped to S and its edges are mapped to pairwise non-crossing circular arcs connecting points in S. We start with a sequence of n^2 points, $q_0, \ldots, q_{n^2-1}$, on $\mathcal{P}$ such that $x_0 \geq 1$ and $x_i \geq 2x_{i-1}$ for $i = 1, \ldots, n^2 - 1$. For our universal point set S, we take the n points $p_i = q_{ni}$ with $i = 0, \ldots, n-1$. We call the points in $q_0, \ldots, q_{n^2-1}$ that are not in the universal point set *helper points*. In the same spirit as Everett et al. [2], we draw G in two steps.

In the first step, we construct a *monotone topological book embedding* of G. This is a drawing of G such that the vertices lie on a horizontal line, called *spine*, and the edges are represented by non-crossing curves, monotonically increasing in the direction of the spine. Di Giacomo et al. [3] have shown that every planar graph has a monotone topological book embedding where each edge crosses the spine exactly once and consists of two semi-circles, one below and one above the spine.

In the second step, we map the vertices of G and the crossings with the spine to the points in $q_0, \ldots, q_{n^2-1}$ in the same order as they appear on the spine of the book embedding, so that the vertices of G are mapped to points in S and the crossings with the spine to helper points. Each edge of G is then drawn as a circular arc that passes through two points of S and a helper point between them. By exploring geometrical properties of circular arcs through three points of $q_0, \ldots, q_{n^2-1}$, we show that the constructed drawing is planar.

Our n-point universal point set uses an area of $2^{O(n^2)}$. Hence, it would be interesting to investigate whether there exist universal point sets of size n (or $o(n^2)$) for circular arc drawings that fit in polynomial area.

The full version of this abstract will appear (under the same title) in *Proc. 25th Canadian Conference on Computational Geometry 2013*, see also http://hal.inria.fr/hal-00846953.

References

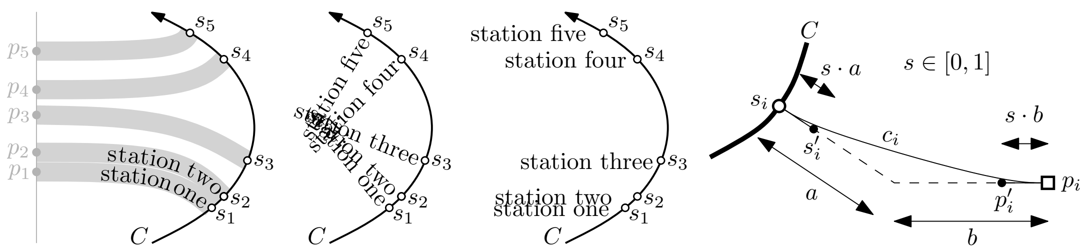
1 Erik. D. Demaine, Joseph. S. B. Mitchell, and Joseph O'Rourke. The open problems project. Website, 2001. URL cs.smith.edu/~orourke/TOPP, accessed May 5, 2012.

2 Hazel Everett, Sylvain Lazard, Giuseppe Liotta, and Stephen Wismath. Universal sets of n points for one-bend drawings of planar graphs with n vertices. *Discrete Comput. Geom.*, 43(2):272–288, 2010.

3 Emilio Di Giacomo, Walter Didimo, Giuseppe Liotta, and Stephen K. Wismath. Curve-constrained drawings of planar graphs. *Comput. Geom. Theory Appl.*, 30:1–23, 2005.

4.5 Labeling Curves with Curved Labels

Jan-Henrik Haunert (Universität Würzburg)
Herman Haverkort (TU Eindhoven)
Benjamin Niedermann (KIT – Karlsruhe Institute of Technology)
Arlind Nocaj (Universität Konstanz)
Aidan Slingsby (City University – London)
Jo Wood (City University – London)



(a) **(b)** **(c)** **(d)**

Figure 5 (a)–(c) Different types of labels for the same metro line C. (a) Curved labels connecting stops of curve C with ports to the left of C. (b) Using straight labels perpendicular to C yields unavoidable clashes. (b) Straight labels can become cluttered when aligned horizontally. (c) Control points of a curved label.

Laying out metro maps automatically is a challenging question in research that in particular comprises finding an appropriate labeling of the metro lines and their stops automatically. We focus on the problem of labeling already embedded metro lines having a smooth curved shape. To the best of our knowledge we present the first considerations towards labeling smooth curved metro lines automatically. So far only labeling of metro lines represented by polygonal chains has been discussed in previous work, for example see [1, 2, 3, 4]. In order to simplify the problem of laying out metro maps with curves we started with only one metro line and assume that its shape is already given by a directed, smooth, non-self-intersecting curve C in the plane, for example described by a Bézier curve. Further, the stops of the metro line are given by n ordered points $s_1, \ldots, s_n$ on C, which we hence call *stops*. Going along C from its beginning to its end we assume that these stops appear in the order $s_1, \ldots, s_n$. Depending on the specific problem formulation one may assume that the stops are not fixed on C, but that we need to find their concrete positions maintaining their order. For each stop s_i we are further given a label that should be placed closely to it. We do not follow traditional map labeling abstracting from the given text by bounding boxes, but we use fat curves prescribing the shape of the labels; see Fig. 5b. We identified the following criteria, which motivated us to consider curved text.

- For aesthetic reasons the labels should integrate into the curved style of the metro map.
- More labels can be placed without intersections when considering curved labels; see Fig. 5a–5c.

Hence, we want to find directed fat curves $c_1, \ldots, c_n$ of fixed width such that for each c_i, $1 \leq i \leq n$ it is true that 1. c_i begins at s_i, 2. c_i does not intersect C, and 3. c_i does not intersect any other curve c_j, $1 \leq j \leq n$. We call these curves *curved labels*. To avoid labels describing wavy lines we assume that $c_1, \ldots, c_n$ are cubic Bézier curves. Thus, we need to specify the four control points s_i, s'_i, p'_i and p_i as depicted in Fig 5d. Using the notion of boundary labeling we call the second endpoint p_i of c_i the *port* of c_i. We considered different choices of those control points, e.g., p_i and s_i are given, s'_i is located on the perpendicular line to C through s_i, p'_i is located on the horizontal line through p_i and the distance of s_i and s'_i is equals to the distance of p_i and p'_i. Thus, in that example only one degree of freedom remains, namely the distance between s_i and s'_i. We think that force-directed methods could be a reasonable way to determine those parameters. However, we still need to identify an appropriate system of forces that also respects conditions 1–3 mentioned above.

To tackle the problem from a more algorithmic point of view we make the assumption that the locations of the stops are given and the ports of the curves are located on a vertical line ℓ that lies to one side of C. To test whether curves $c_1, \ldots, c_n$ exist with respect to condition 1–3, we developed basic algorithms that either follow a greedy strategy or a dynamic-programming approach. For the greedy strategy we assume that the number of stops and ports coincide. Thus, going along ℓ from bottom to top the assignment between stops and ports is unique. If the number of ports is greater than the number of stops we apply a dynamic-programming approach using the observation that a curve c_i of a solution partitions the instance into two independent sub-instances. We think that these algorithms can be used for finding an initial solution, which later can be refined with a force-directed algorithm as described above.

For future work we identified the following questions to be answered. How can the approaches be extended if ports lie on both sides of the given metro line C? Where should ports be placed? How can the case be handled that several metro lines are given? How to define an appropriate system of forces in order to fine-tune the parameters of the Bézier curves?

References

1. Maria Angeles Garrido, Claudia Iturriaga, Alberto Márquez, José Ramón Portillo, Pedro Reyes, and Alexander Wolff. Labeling subway lines. In Peter Eades and Tadao Takaoka, editors, *ISAAC*, volume 2223 of *Lecture Notes in Computer Science*, pages 649–659. Springer, 2001.
2. Martin Nöllenburg and Alexander Wolff. Drawing and labeling high-quality metro maps by mixed-integer programming. *IEEE Transactions on Visualization and Computer Graphics*, 17(5):626–641, May 2011.
3. Jonathan Stott, Peter Rodgers, Juan Carlos Martinez-Ovando, and Stephen G. Walker. Automatic metro map layout using multicriteria optimization. *IEEE Transactions on Visualization and Computer Graphics*, 17(1):101–114, January 2011.
4. Alexander Wolff. Drawing subway maps: A survey. *Inform., Forsch. Entwickl.*, 22(1):23–44, 2007.

4.6 Graphs with Circular Arc Contact Representation

David Eppstein (University California – Irvine)
Éric Fusy (Ecole Polytechnique – Palaiseau)
Stephen Kobourov (University of Arizona – Tucson)
André Schulz (Universität Münster)
Torsten Ueckerdt (University of California – Santa Barbara)

An *arc* is an open connected subset of a circle. For an arc a the two points in the closure of a that are not on a are called its endpoints. If a has only one endpoint, i.e., it is a circle with one point removed, then this endpoint is counted with multiplicity two. Let $\mathcal{A}$ be an arrangement of disjoint circular arcs in the plane. We say that an arc a_1 touches an arc a_2, if one of the endpoints of a_1 lies on a_2. A graph G is represented by $\mathcal{A}$ if there is a bijection between the vertices of G and the arcs in $\mathcal{A}$ with the property that two arcs touch in the arrangement exactly if their corresponding vertices are adjacent in G. We call $\mathcal{A}$ a *circular arc contact representation* of G, *arc-representation* for short. Notice that G might contain parallel edges but no self-loops. Fig. 6 shows an example of a circular arc arrangement and its associated graph.

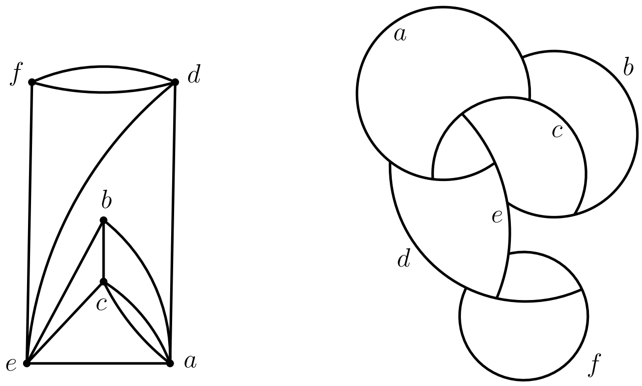


Figure 6 A graph (on the left) and one of its arc-representation (on the right).

We study the question of which graphs can be represented by an arc-representation. In every arc-representation each arc touches at most two other arcs with its endpoints. Hence, there are at most twice as many edges in G as there are vertices. This observation is not only true for the full arrangement, but also for every subset of it. Furthermore, since the arrangement is non-intersecting, the graph G has to be planar as well. In terms of the following notation, graphs with arc-representation are $(2,0)$-sparse and planar.

▶ **Definition 1** (Lee and Streinu [3]). A graph $G = (V, E)$ is called (k,ℓ)-*tight* for some natural numbers k, ℓ, if

1. $|E| = k|V| - \ell$,
2. for every subset $X \subseteq V$ with induced edge set $E[X]$ we have that $|E[X]| \leq k|V| - \ell$.

If only condition 2. is fulfilled we call the graph (k,ℓ)-*sparse*.

Every planar $(2,3)$-sparse graph has contact representation with straight line segments [1], and therefore an arc-representation. On the other hand, every graph with a segment contact representation is necessarily $(2,3)$-sparse. We claim the following:

▶ **Theorem 2.** *Every planar* $(2, 1)$*-sparse graph has an arc-representation.*

Since graphs with arc-representations are closed under taking subgraphs, it clearly suffices to prove Theorem 2 for $(2, 1)$-tight graphs. Our main tool for the proof is a construction sequence that produces G by sequentially adding a vertex with low vertex degree, while maintaining the planarity and the tightness constraint of the graph. These local operations are called *Henneberg steps* [2]. When introducing a vertex of degree $k + 1$, we have to add $k - 1$ edges, such that tightness constraint remains fulfilled. If the new vertex has degree $k + 1$ the modification is called a Henneberg-k step, Hk step for short. In general Henneberg steps do not have to preserve planarity. However, in our setting it suffices to consider planarity-preserving *planar* Henneberg steps.

Every planar $(2, 2)$-tight graph can be constructed by a series of planar H1 and H2 steps starting from a graph with two vertices and a double edge. Similarly, every planar $(2, 1)$-tight graph can be constructed by a series of planar H1 and H2 steps starting from a graph with two vertices and a triple edge. In the $(2, 0)$-tight case we have to allow three Henneberg steps, H1, H2 and H3. Here the base case is any plane graph each of whose components consists of two vertices and 4 parallel edges.

To show that every planar $(2, 1)$-tight graph G has an arc-representation we follow its construction sequence. We start with an arc-representation of the triple edge and then apply the Henneberg steps that lead to G. All these steps are carried out geometrically in the arc-representation. This way we finally construct an arc-representation of G. The same strategy can be used to show that all $(2, 2)$-tight graphs have an arc-representation. The only difference is the base case.

To carry out the Henneberg steps in the arc-representation we need one more tool. When adding an arc to the arc-representation we have to ensure that we can establish the desired connectivity between the arcs. To guarantee this we enhance the arc-representation with so called *corridors* as spatial certificates for the *circular arc visibility*. We can show that each Henneberg step in the construction sequence maintains a required set of corridors. Some more technical parts of the proposed proof still need to be checked.

Our techniques do not apply for general planar $(2, 0)$-tight graphs yet. The main difference to the previous cases is that during the construction the graph might become disconnected. However, if the construction sequence maintains the connectivity we can prove that the graph has an arc-representation.

References

1 Muhammad Jawaherul Alam, Therese Biedl, Stefan Felsner, Michael Kaufmann, and Stephen G. Kobourov. Proportional contact representations of planar graphs. In *Proc. 19th Symposium on Graph Drawing (GD '11)*, volume 7034 of *LNCS*, pages 26–38. Springer, 2012.

2 Lebrecht Henneberg. *Die graphische Statik der starren Systeme.* B.G. Teubner, Leipzig, 1911.

3 Audrey Lee and Ileana Streinu. Pebble game algorithms and sparse graphs. *Discrete Mathematics*, 308(8):1425–1437, 2008.

5 Open Problems

Apart from the six problems, for which working groups formed during the seminar, we collected several additional open problems during the two open problem sessions. They are briefly summarized in this section.

5.1 Drawing r-partite hypergraphs

Günter Rote (FU Berlin)

A 3-uniform 3-partite hypergraph is a subset of $A \times B \times C$ for three disjoint sets A, B, C. It can be drawn by placing points for the vertices in A, B, C on three vertical lines and drawing each triplet (hyperedge) (a, b, c) as a quadratic function (parabolic arc) joining the three corresponding points. The hypergraph can, in principle, be reconstructed uniquely from this drawing, even if different arcs going through a vertex have the same tangent direction. The open question is to define quality criteria for the visual appearance, beauty, clarity, or clutteredness of such a drawing and to see whether one can optimize the drawing by placing the vertices suitably. The problem extends to r-uniform r-partite hypergraphs by using curves of degree $r - 1$.

5.2 Characterization of Planar Lombardi Graphs

David Eppstein (UC Irvine)

Lombardi drawings are drawings of graphs that use circular arcs to represent edges and require that every vertex v has perfect angular resolution $2\pi/\deg(v)$, see Section 3.1.8 and [1]. A graph that admits a planar Lombardi drawing is called a planar Lombardi graph. Not every planar graph has a planar Lombardi drawing [12] and it is an open question to characterize the planar Lombardi graphs. In particular, it is unknown whether every outerplanar graph has a planar Lombardi drawing.

References

1 Christian A. Duncan, David Eppstein, Michael T. Goodrich, Stephen G. Kobourov, and Martin Nöllenburg. Lombardi drawings of graphs. *J. Graph Algorithms and Applications*, 16(1):85–108, 2012.

5.3 Small Path Covers in Planar Graphs

André Schulz (Uni Münster)

Let $G = (V, E)$ be a graph of a graph class $\mathcal{G}$ with $|E| = m$ edges. A *path-cover* $\mathcal{P} = \{P_1, \ldots, P_k\}$ is a partition of E into edge-disjoint simple paths. The size of the cover is $\sigma(\mathcal{P}) = k$. I am interested in upper and lower bounds for the quantity

$$\mathrm{pc}(G) := \min_{G \in \mathcal{G}} \max_{\mathcal{P} \text{ is path-cover of } G} m/\sigma(\mathcal{P}).$$

In other words, how large is the average path length in the smallest path-cover in the worst case. The graphs classes $\mathcal{G}$ I am interested in are (1) planar 3-connected graphs, and (2) triangulations.

There is a simple observation for an upper bound: In every odd-degree vertex one path has to start/end. By Euler's formula a planar graph can have up to $3|V| - 6$ edges. So when all vertices in a triangulation G have odd degree then $\mathrm{pc}(G) \leq 6 - \varepsilon$. The variable $\varepsilon > 0$ can be made arbitrarily small by considering larger triangulations. The same bound follows from the fact that $\lceil(|V|-1)/2\rceil$ paths have to pass through a vertex of degree $|V|-1$.

▶ Problem 1. Does any 3-connected planar graph (triangulation) $G=(V,E)$ have a path-cover of size $\mathrm{pc}(G) \geq 6 - c/|V|$, for $c > 0$ being a constant.

The problem might be related to the *linear arboricity conjecture*. This conjecture claims that the number of linear forests (disjoint unions of paths) of any graph is either $\lceil\Delta/2\rceil$ or $\lceil(\Delta+1)/2\rceil$, where Δ denotes the maximal vertex degree of the graph. It was proven for planar graphs by Wu [3].

If the graph is a triangulation it can be decomposed into edge-disjoint simple paths that have all exactly three edges [2]. The same is true for cubic bridge-less graphs [1]. So in both cases we have a lower bound of $\mathrm{pc}(G) \geq 3$ for those graph classes.

▶ Problem 2. Does there exist for every planar 3-connected graph a path-cover in which every path has exactly three edges, with the exception of one path that might be shorter?

These questions are motivated from proving lower bounds for certain problems in graph drawing with circular arcs.

References

1 André Bouchet and Jean-Luc Fouquet. Trois types de décompositions d'un graphe en chaînes. In C. Berge, D. Bresson, P. Camion, J. Maurras, and F. Sterboul, editors, *Combinatorial Mathematics Proceedings of the International Colloquium on Graph Theory and Combinatorics*, volume 75, pages 131 – 141. North-Holland, 1983.
2 Roland Häggkvist and Robert Johansson. A note on edge-decompositions of planar graphs. *Discrete Mathematics*, 283(1-3):263–266, 2004.
3 Jian Liang Wu. On the linear arboricity of planar graphs. *J. Graph Theory* 31:129–134, 1999.

5.4 Self-approaching Networks on Planar Point Sets

Fabrizio Frati (University of Sydney)

A curve C in the plane with end-points u and v is called self-approaching from u to v if, for every three points a, b, and c in this order along C from u to v, the Euclidean distance between a and c is greater than the Euclidean distance between b and c. A drawing D of a graph G is called self-approaching if, for every ordered pair (u, v) of vertices of G, there exists a path in G whose drawing in D is a self-approaching curve from u to v.

The problem asks whether, for every point set P in the plane, there exists a planar drawing D of a graph G that spans the points in P and that is self-approaching. The problem has been defined by Alamdari et al. [1].

References

1 Soroush Alamdari, Timothy M. Chan, Elyot Grant, Anna Lubiw, and Vinayak Pathak. Self-approaching Graphs. In *Proc. 20th Symposium on Graph Drawing (GD '12)*, volume 7704 of *LNCS*, pages 260–271. Springer, 2013.

5.5 Improving Curved Drawings with Edge Direction and Curvature Optimization

Kai Xu (Middlesex University)

We presented three related problems on improving Lombardi-like drawings:

1. The first is how to choose the edge curve direction to improve the layout. Different criteria can be applied here. Examples are to reduce edge crossing or improve the continuity of paths between nodes.
2. The second one is weighting between angular resolution and other aesthetics criteria. A method that considers different criteria may produce a layout that is better than a method that only focuses on angular resolution. The research question is how to find the right balance, which may vary from graph to graph.
3. The last one is to have both straight and curved edges in a drawing. Curved edges are only used if they can improve certain aesthetic metrics, for example less edge crossings. The research questions are when to use curved edges and how to do this efficiently.

5.6 Improving Graph Readability by Spatial Distortion of Node-Link-Based Graph Depictions within Geographical Contexts

Danny Holten (SynerScope BV)

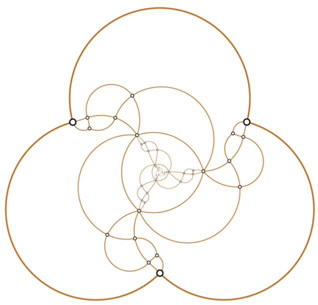
A multitude of layout algorithms exists for the generation of 2D/3D node-link-based graph layouts. Most of these algorithms are fairly unrestricted in the sense that they have a high degree of freedom with respect to the final placement of nodes and/or links within the 2D/3D space into which the layout is embedded. However, there are situations in which additional and sometimes stringent restrictions are applicable to (mostly) node placement, which can severely limit the generation of an optimized and easily readable layout due to the presence of node clutter. A typical example is when node positions are determined by, e.g., locations on a geographical map; this does not really allow for optimized placement of nodes for improved readability through uncluttered positioning. In such cases, (mild) continuous distortions can be applied to the layout to unclutter the graph and, hence, improve readability by utilizing spatial graph sparsity as a measure, e.g., by locally contracting (parts of) the graph where node/link density is low and locally expanding (parts of) the graph where node/link density is high.

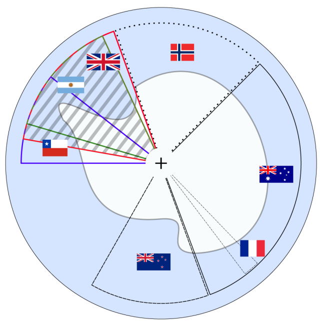
6 Exhibition: Bending Reality

Maxwell J. Roberts (University of Essex)

Previous seminars concerned with graph drawing and mapping have been staged in conjunction with successful exhibitions which have highlighted the visual nature of the subject matter [10461 – Schematization in Cartography, Visualization, and Computational Geometry – Underground Maps Unravelled; 12261 – Putting Data on the Map – Beyond the Landscape]. Continuing with this tradition, an exhibition was staged in conjunction with this seminar: *Bending Reality: Where Arc and Science Meet*.

The exhibition comprised works directly relevant to the theme of the seminar, demonstrating the use of smooth curves of any kind in order to depict spatial or abstract relationships in the arts and science. The intention was that by showcasing relevant drawings and other visual works this would provide an attractive environment in which to work, and stimulate discussion amongst seminar participants. Exhibits were solicited from seminar members and were reviewed by the exhibition committee. Part of the seminar program comprised an unofficial opening, in which participants toured the exhibits and artists described their own works. In the meantime, the exhibition is available online. Please visit **http://www.dagstuhl.de/ueber-dagstuhl/kunst/13151**.





Exhibition Synopsis

Nature is curved. From tangled branches to meandering rivers and irregular coastlines, there is barely a straight line to be seen. In centuries past, information designers have sought to emulate this, with tree-like diagrams and twisting connections, but others have objected to the inefficiency of curves. Straight lines are direct, simple and easy to follow, so why take pointless diversions?

Straight lines can play a part in simplifying the world, highlighting routes and connections, but they can only go so far, and sometimes curves may be better suited. Whether these are free-form Béziers or regular geometric patterns, in some circumstances they can be more effective or more aesthetic, and the task of the information designer is to take a flexible approach, identifying the best tools and techniques for the particular task to hand.

This exhibition presents information from a variety of contexts, the linking theme is that the designers have investigated the use of curves in order to show routes, structures, movement, time and connections.

The exhibition was divided into seven sections, as described below. It has now been implemented online, and those artworks that are now part of the online exhibition are listed. All listed exhibits were contributed by seminar participants.

6.1 Curved Annotations of the World

The world is a sphere, but even when it is shown as a flat projection, curves are often used, especially to show movement.

- The German war society 1939 to 1945. Exploitation, interpretations, exclusion. Andreas Reimer. 2005
- The Flow of Whisky. Kevin Buchin, Bettina Speckmann, and Kevin Verbeek. 2010
- A Bicycle's Journey. Jo Wood. 2013
- Airline graph after several steps of an edge bundling algorithm. Emden Gansner, Yifan Hu, Stephen North and Carlos Scheidegger. 2011

6.2 Curving the World

Curves in the world are rarely simple, whether twisted gnarled branches or a fractal coastline. Even man-made structures such as railway lines can have complex, uncertain trajectories. These can be smoothed and simplified, but sometimes there is a temptation to go further, turning the world into alien geometric shapes.

- O-ZONE III. Arthur van Goethem, Wouter Meulemans, Andreas Reimer, Herman Haverkort, and Bettina Speckmann. 2013
- Curve Limit IV. Arthur van Goethem, Wouter Meulemans, Bettina Speckmann, and Jo Wood. 2013
- Countries in Southeast Asia, with straight edges, Bézier curves, and circular arcs. Arthur van Goethem, Wouter Meulemans, Andreas Reimer, Herman Haverkort, and Bettina Speckmann. 2013
- Circular-arc cartograms of the states of Germany. Martin Nöllenburg. 2013

6.3 Early Metro Maps

The most widely used examples of the curves of reality being converted into straight lines are metro maps worldwide. This information design technique is relatively recent, first used in Berlin (1931) and London (1933). Before this, curves on maps generally were used in an attempt to represent reality. However, there are also examples of attempts to simplify, even going as far as using a regular circle.

- London Underground map based upon designs by F.H. Stingemore, 1925-1932. Maxwell J. Roberts. 2009
- Metropolitan Line carriage diagram from the early 1920s. Maxwell J. Roberts. 2012
- Berlin S-Bahn map based upon a 1931 design. Maxwell J. Roberts. 2012
- London Underground map based upon a 1933 design by Henry Beck. Maxwell J. Roberts. 2009
- Paris Metro map based upon a 1939 design by Loterie Nationale, France. Maxwell J. Roberts. 2012
- Moscow Metro schema based upon 1970s designs. Maxwell J. Roberts. 2009

6.4 Metro Maps Using Freeform Béziers

Sometimes, straight lines fail to simplify reality. In the case of a complex highly interconnected network, gentle curves smooth away harsh zigzags, potentially revealing the underlying structure of the networks.

- The Madrid Metro: An all-curves design. Maxwell J. Roberts. 2009
- Berlin all-curves U- and S-Bahn network map. Maxwell J. Roberts. 2012
- Automatically Generated Drawing of the London Underground using Bézier curves. M. Fink, H. Haverkort, M. Nöllenburg, M. Roberts, J. Schuhmann, and A. Wolff. 2013
- An all-curves map of the Paris Metro. Maxwell J. Roberts. 2007
- Curvy tube map. Maxwell J. Roberts. 2008
- Subway network of Vienna drawn by a force-Directed method using Bézier curves. A. Wolff, M. Fink, H. Haverkort, M. Nöllenburg, and J. Schuhmann. 2013

6.5 Metro Maps Using Concentric Circles

Highly abstract and stylised, this new way of showing networks does not necessarily offer simplified line trajectories, but nonetheless presents a highly organised view of the network, which many people find striking.

- The Vienna U-Bahn in minimalist style. Therese Biedl and Maxwell J. Roberts. 2012
- The Madrid Metro: A concentric circles design. Maxwell J. Roberts. 2013
- A concentric circles map of the Moscow Metro. Maxwell J. Roberts. 2013
- Berlin concentric circles U- and S-Bahn network map. Maxwell J. Roberts. 2013
- A map of the London Underground based upon concentric circles, spokes, and tangents. Smooth corners version. Maxwell J. Roberts. 2013

6.6 Curved Relationships

In theory, the display of abstract concepts such as time and relatedness need not be constrained by preconceptions derived from our experience of nature. In practice, use of curves can add an aesthetic dimension, or even assist in the presentation of information.

- Mark Lombardi: Kunst und Konspiration. Film poster. 2012
- The collaboration network between Jazz bands. S. Pupyrev, L. Nachmanson, S. Bereg, and A.E. Holroyd. 2011
- 40 Jahre Universität Konstanz. Ulrik Brandes. 2006
- Poster for 21st International Symposium on Graph Drawing. David Auber. 2013
- Spaghetti à la Wehrli. Mereke van Garderen and Bettina Speckmann. 2013

6.7 Mathematical Abstractions

Curves are no stranger to the mathematical world, and sometimes enable combinatorial structures, abstract graphs and other concepts to be visualised more efficiently than with straight lines.

- Iterated Möbius-transformations. André Schulz. 2013
- Random confluent Hasse diagram on 222 points. David Eppstein. 2013

- Flowerpot. Günter Rote. 2013
- Lombardi drawings. C. Duncan, D. Eppstein, M. Goodrich, S. Kobourov, and M. Löffler. 2011
- The Petersen family. David Eppstein. 2010
- Lombardi drawing of the full binary outertree on 1025 vertices (in a non-standard embedding). Maarten Löffler. 2012
- A selection of graphs from the Rome library drawn using a Lombardi force-directed algorithm. Stephen Kobourov. 2011
- Hyperbolic tessellation generated by a Fuchsian group with genus 1 and a period of 3. Jakob von Raumer. 2013
- Covering of the hyperbolic plane. Mikhail Bogdanov, Olivier Devillers, and Monique Teillaud. 2013
- The Nauru graph in 3d. David Eppstein. 2008

Participants

- Patrizio Angelini
Università di Roma III, IT
- Michael Bekos
National TU – Athens, GR
- David Eppstein
Univ. of California – Irvine, US
- Fabrizio Frati
The University of Sydney, AU
- Eric Fusy
Ecole Polytechnique – Palaiseau, FR
- Martin Gronemann
Universität Köln, DE
- Jan-Henrik Haunert
Universität Würzburg, DE
- Herman J. Haverkort
TU Eindhoven, NL
- Michael Hemmer
TU Braunschweig, DE
- Danny Holten
SynerScope BV, NL
- Michael Kaufmann
Universität Tübingen, DE
- Stephen G. Kobourov
Univ. of Arizona – Tucson, US
- Sylvain Lazard
INRIA Grand Est – Nancy, FR
- Maarten Löffler
Utrecht University, NL
- Tamara Mchedlidze
KIT – Karlsruhe Institute of Technology, DE
- Wouter Meulemans
TU Eindhoven, NL
- Lev Nachmanson
Microsoft Corp. – Redmond, US
- Benjamin Niedermann
KIT – Karlsruhe Institute of Technology, DE
- Arlind Nocaj
Universität Konstanz, DE)
- Martin Nöllenburg
KIT – Karlsruhe Institute of Technology, DE
- Sergey Pupyrev
Univ. of Arizona – Tucson, US
- Helen C. Purchase
University of Glasgow, GB
- Andreas Reimer
Universität Heidelberg, DE
- Maxwell J. Roberts
University of Essex, GB
- Günter Rote
FU Berlin, DE
- André Schulz
Universität Münster, DE
- Aidan Slingsby
City University – London, GB
- Bettina Speckmann
TU Eindhoven, NL
- Monique Teillaud
INRIA Sophia Antipolis – Méditerranée, FR
- Torsten Ueckerdt
KIT – Karlsruhe Institute of Technology, DE
- Kevin Verbeek
University of California – Santa Barbara, US
- Alexander Wolff
Universität Würzburg, DE
- Jo Wood
City University – London, GB
- Kai Xu
Middlesex University, GB