

Poster Abstracts

Visual Analysis of Temporal Fiscal Networks with TeFNet

Walter Didimo, Luca Grilli, Giuseppe Liotta, Fabrizio Montecchiani,
and Daniele Pagliuca^(✉)

Università degli Studi di Perugia, Perugia, Italy
{walter.didimo,luca.grilli,giuseppe.liotta,
fabrizio.montecchiani}@unipg.it, daniele.pagliuca@studenti.unipg.it

The development of software systems for the analysis of economic and financial networks is a fundamental activity to contrast tax evasion, fiscal frauds, and money laundering phenomena (see, e.g., [3, 6, 8]). Of particular interest in this context is the design of visual analytics systems (see, e.g., [1, 4, 5, 7]).

In this poster, we present TEFNET, a new system specifically designed to support public officers in tax evasion discovery and risk analysis. It is an evolution of TAXNET [4]: A visual analytics decision support system for tax evasion discovery. TEFNET inherits all the TAXNET’s functionalities while overcoming its main limitation: The lack of a native support for temporal queries and visualizations. TEFNET’s main ingredients are: (i) a network data model to represent time-varying relationships between taxpayers, called *temporal fiscal network*; (ii) a visual query language to easily define and search for suspicious (time-dependent) patterns in a temporal fiscal network; (iii) visualization functionalities to interactively explore the subgraphs that match a pattern. Both the visual query language and the graph visualization techniques rely on a suitable *timeline* approach [2, 9], which maps the time dimension to a space dimension.

Temporal Fiscal Networks. A *temporal fiscal network* is a directed graph G , whose nodes represent taxpayers (persons or companies), and whose edges represent oriented relationships between pairs of taxpayers, such as economic transactions, shareholdings and legal acts. Each element (node or edge) of G exists in a specific time interval (the *validity period*) going *from* an initial date *to* an ending date. In addition, an element can have one or more associated attributes, which can be *static* (time-independent), *temporal* (time-dependent), or *periodical*, i.e., time-dependent according to fiscal or business calendar periods.

Visual Query Language. The visual query language of TEFNET allows the user to define time-dependent patterns to be matched in G . A *pattern* p is a pair $\langle G_p, R_p \rangle$, where G_p is a graph that defines the topology of p , and R_p is a set of rules on the nodes and on the edges of G_p . The user can restrict the analysis to data within a desired *time range* and specify a *time slicing unit* (e.g., year or month) to partition the time range into intervals (*slices*) of the same length.

Work in cooperation with the Italian Revenue Agency (IRV). We thank in particular Carlo Palumbo, Mario Landolfi, and Giuseppe De Luca for their support.

Research supported in part by the project: “Algoritmi e sistemi di analisi visuale di reti complesse e di grandi dimensioni” - Ricerca di Base 2018, Dipartimento di Ingegneria dell’Università degli Studi di Perugia.

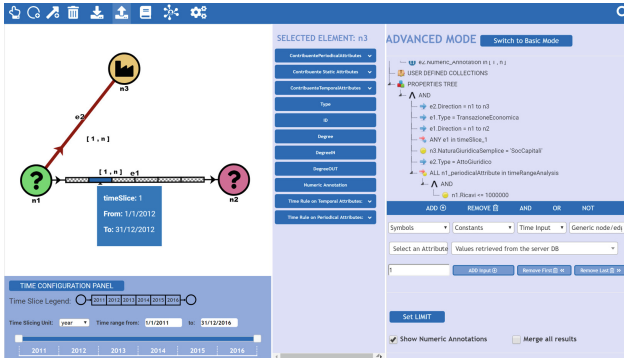


Fig. 1. Visual query language interface of TEFNET for graph pattern definition.

She can also define, in a specific time slice or in the time range, temporal rules on nodes, edges and related attributes. For example, in the pattern of Fig. 1, the timeline edge e_1 is used to express the presence of economic transactions from a node n_1 to a node n_2 in the year 2012. This rule is defined by means of the *quantity operator* ANY and it is visually conveyed in G_p by a solid filling of the corresponding slice. Other quantity operators for a slice are SINGLE (diagonal lines) - only one relation in that slice and NONE (white) - no relation in that slice. A slice without quantity operators is visually conveyed with a chess filling.

Visual Exploration. In response to a user query, TEFNET returns all the subgraphs that match the specified pattern. The analysis of a result is performed through interactive visual exploration. Edges can be visually displayed in a standard mode or as timeline edges. A timeline edge is visually split into slices as in the query interface. Each slice is filled with a color whose intensity is proportional to the value of some desired function, which may represent the weight of the edge, e.g., the amount of the transaction, or the presence/absence of relations. This makes it possible for the user to easily capture in a unique view the trend of a specific parameter over the time range of analysis. The user can also expand the analysis of a result by introducing other neighbors in the current visualization. For example, Fig. 2 shows a visualization of a subgraph after some exploration steps, starting from a result of the pattern defined in Fig. 1.

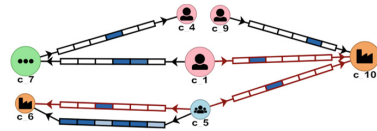


Fig. 2. Visualization of a subgraph after some exploration steps.

We tested TEFNET in a real working environment on a real 3-year fiscal network of approximately 800 K nodes and 1.9M edges. The experimental tasks were performed by expert tax officers, who were asked to find subjects having specific time-varying relations with a given taxpayer. The results show that using TEFNET may significantly improve time and accuracy of the analysis at the IRV.

References

1. Argyriou, E.N., Symvonis, A., Vassiliou, V.: A fraud detection visualization system utilizing radial drawings and heat-maps. In: Laramee, R.S., Kerren, A., Braz, J. (eds.) IVAPP 2014. pp. 153–160. SciTePress (2014). <https://doi.org/10.5220/0004735501530160>
2. Beck, F., Burch, M., Diehl, S., Weiskopf, D.: A taxonomy and survey of dynamic graph visualization. *Comput. Graph. Forum* **36**(1), 133–159 (2017). <https://doi.org/10.1111/cgf.12791>
3. Colladon, A.F., Remondi, E.: Using social network analysis to prevent money laundering. *Expert Syst. Appl.* **67**, 49–58 (2017). <https://doi.org/10.1016/j.eswa.2016.09.029>
4. Didimo, W., Giamminonni, L., Liotta, G., Montecchiani, F., Pagliuca, D.: A visual analytics system to support tax evasion discovery. *Decis. Support Syst.* **110**, 71–83 (2018). <https://doi.org/10.1016/j.dss.2018.03.008>
5. Didimo, W., Liotta, G., Montecchiani, F.: Network visualization for financial crime detection. *J. Vis. Lang. Comput.* **25**(4), 433–451 (2014). <https://doi.org/10.1016/j.jvlc.2014.01.002>
6. González, P.C., Velásquez, J.D.: Characterization and detection of taxpayers with false invoices using data mining techniques. *Expert Syst. Appl.* **40**(5), 1427–1436 (2013). <https://doi.org/10.1016/j.eswa.2012.08.051>
7. Huang, M.L., Liang, J., Nguyen, Q.V.: A visualization approach for frauds detection in financial market. In: Banissi, E., Stuart, L.J., Wyeld, T.G., Jern, M., Andrienko, G.L., Memon, N., Alhajj, R., Burkhard, R.A., Grinstein, G.G., Groth, D.P., Ursyn, A., Johansson, J., Forsell, C., Cvek, U., Trutschl, M., Marchese, F.T., Maple, C., Cowell, A.J., Moere, A.V. (eds.) IV 2009. pp. 197–202. IEEE (2009). <https://doi.org/10.1109/IV.2009.23https://doi.org/10.1109/IV.2009.23>
8. Ngai, E.W.T., Hu, Y., Wong, Y.H., Chen, Y., Sun, X.: The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decis. Support Syst.* **50**(3), 559–569 (2011). <https://doi.org/10.1016/j.dss.2010.08.006>
9. Schmauder, H., Burch, M., Weiskopf, D.: Visualizing dynamic weighted digraphs with partial links. In: Braz, J., Kerren, A., Linsen, L. (eds.) IVAPP 2015 - Proceedings of the 6th International Conference on Information Visualization Theory and Applications, Berlin, Germany, 11–14 March, 2015. pp. 123–130 (2015). <https://doi.org/10.5220/0005303801230130>

Multilevel Planarity

Lukas Barth^(✉), Guido Brückner, Paul Jungeblut, and Marcel Radermacher

Department of Computer Science,
Karlsruhe Institute of Technology, Karlsruhe, Germany
{lukas.barth,brueckner,radermacher}@kit.edu,
paul.jungeblut@student.kit.edu

An upward-planar drawing is a planar drawing where each edge is drawn as a strictly y -monotone curve. While testing upward planarity of a graph is an NP-complete problem in general [11], efficient algorithms are known for single-source graphs and for embedded graphs [5, 6]. One notable specialization of upward planarity is that of level planarity. A level graph is a directed graph $G = (V, E)$ together with a level assignment $\gamma : V \rightarrow \mathbb{Z}$ that assigns an integer level to each vertex and satisfies $\gamma(u) < \gamma(v)$ for all $(u, v) \in E$. A drawing of G is level planar if it is upward planar, and for the y -coordinate of each vertex $v \in V$ it holds that $y(v) = \gamma(v)$. Level-planarity testing and embedding is feasible in linear time for single-source graphs and graphs with multiple sources, the latter case being considerably more complex [9, 13]. There exist further level-planarity variants on the cylinder and on the torus [1, 3] and there has been considerable research on further-constrained versions of level planarity [2, 7, 10, 12, 14].

We introduce and study the *multilevel-planarity testing* (MLPT) problem, which is a generalization of upward planarity and level planarity. Let $G = (V, E)$ be a directed graph and let $\ell : V \rightarrow \mathcal{P}(\mathbb{Z})$ be a function that assigns a finite set of integers to each vertex. A multilevel-planar drawing of G is an upward planar drawing of G such that the y -coordinate of each vertex $v \in V$ satisfies $y(v) \in \ell(v)$.

We present linear-time algorithms for testing multilevel planarity of embedded graphs with a single source (sT -graphs) and for oriented cycles. To this end, we characterize multilevel-planar sT -graphs as subgraphs of certain planar graphs with a single source and a single sink (st -graphs). Similar characterizations exist for upward planarity and level planarity [9, 15]. The idea behind our characterization is that we can insert edges into any given multilevel-planar drawing of a graph so as to make it an st -graph while maintaining multilevel planarity. This technique is similar to the one found by Bertolazzi et al. [6] for upward planarity, and in fact is built on top of it. For the obtained st -graphs, we may assume without loss of generality that the multilevel assignment ℓ has *normal form*, i.e., for all $(u, v) \in E$ it is $\min \ell(u) < \min \ell(v)$ and $\max \ell(u) < \max \ell(v)$. Then, we can test multilevel planarity by greedily attempting to place the vertices of G in topological order on the lowest possible level. For oriented cycles, we identify sets of vertices of minimal cardinality that have to be placed on the lowest and highest possible levels. Assuming a multilevel-planar drawing exists, the remaining vertices can then be placed greedily as low as possible between them. Both algorithms test multilevel planarity in linear time and generate a multilevel-planar drawing within the same running time.

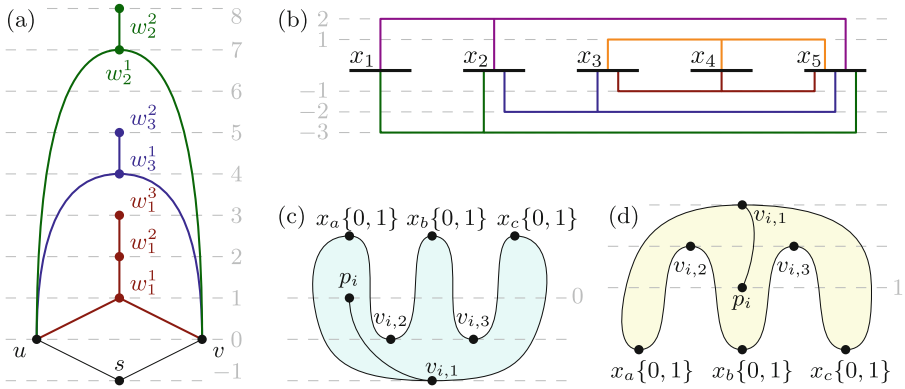


Fig. 1. (a) The *sT*-graph formed by three task gadgets (red, green and blue sub-graphs). (b) A rectilinear embedding of a planar monotone 3-SAT instance. E-shapes above/below the variables are clauses containing only negative/positive literals. (c) The gadget for a positive clause. (d) The gadget for a negative clause.

Complementing these algorithmic results, we show that MLPT is NP-complete even in very restricted cases, namely for *sT*-graphs without a fixed embedding, for trees and for general embedded *ST*-graphs (graphs with multiple sources and sinks). This contrasts both the upward planar and level planar setting, where these problems are solvable in polynomial time (see Table 1 for a full comparison).

The first two reductions use the strongly NP-complete single-processor scheduling problem with individual release times, deadlines and processing times. For a set of tasks there exists a non-preemptive one-processor schedule if and only if a crossing free nesting of the task gadgets in the *sT*-graph shown in Fig. 1(a) exists. Using similar gadgets, this *sT*-graph can be transformed into a tree. Here the release time and deadline of a task define the interval of possible levels for each of the gadgets vertices. The number of vertices in the gadget is the processing time.

To show NP-completeness for embedded *ST*-graphs, we give a polynomial reduction from planar monotone 3-SAT [8]. Given a rectilinear embedding of the variables and clauses as in Fig. 1(b), we substitute the E-shaped clauses by the gadgets shown in Fig. 1(c,d). Now there is a multilevel-planar drawing, if and only if there is a truth assignment of the planar monotone 3-SAT instance.

Table 1. Result overview

	Not embedded		Fixed combinatorial embedding		
	Trees	<i>sT</i> -graph	Cycle	<i>sT</i> -graph	<i>ST</i> -graph
Upward planarity	$O(1)$ [4]	$O(n)$ [6]	$O(n)$ [5]	$O(n)$ [5]	P [5]
Multilevel planarity	NPC	NPC	$O(n)$	$O(n)$	NPC
Level planarity	$O(n)$ [13]	$O(n)$ [13]	$O(n)$ [13]	$O(1)$ [13]	?

References

1. Angelini, P., Da Lozzo, G., Di Battista, G., Frati, F., Patrignani, M., Rutter, I.: Beyond level planarity. In: Hu, Y., Nöllenburg, M. (eds.) GD 2016. LNCS, vol. 9801, pp. 482–495. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-50106-2_37
2. Angelini, P., Da Lozzo, G., Di Battista, G., Frati, F., Roselli, V.: The importance of being proper (in clustered-level planarity and t -level planarity). *Theor. Comput. Sci.* **571**, 1–9 (2015)
3. Bachmaier, C., Brandenburg, F.J., Forster, M.: Radial level planarity testing and embedding in linear time. *J. Graph Algorithms Appl.* **9**(1), 53–97 (2005)
4. Battista, G.D., Eades, P., Tamassia, R., Tollis, I.G.: *Graph Drawing: Algorithms for the Visualization of Graphs*. 1st edn. Prentice Hall PTR (1998)
5. Bertolazzi, P., Di Battista, G., Liotta, G., Mannino, C.: Upward drawings of tri-connected digraphs. *Algorithmica* **12**(6), 476–497 (1994)
6. Bertolazzi, P., Di Battista, G., Mannino, C., Tamassia, R.: Optimal upward planarity testing of single-source digraphs. *SIAM J. Comput.* **27**(1), 132–169 (1998)
7. Brückner, G., Rutter, I.: Partial and constrained level planarity. In: Klein, P.N. (ed.) *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2017)*, pp. 2000–2011 (2017)
8. De Berg, M., Khosravi, A.: Optimal binary space partitions for segments in the plane. *Int. J. Comput. Geom. Appl.* **22**(3), 187–205 (2012)
9. Di Battista, G., Tamassia, R.: Algorithms for plane representations of acyclic digraphs. *Theor. Comput. Sci.* **61**(2), 175–198 (1988)
10. Forster, M., Bachmaier, C.: Clustered level planarity. In: Van Emde Boas, P., Pokorný, J., Bieliková, M., Štuller, J. (eds.) *SOFSEM 2004*. LNCS, vol. 2932, pp. 218–228. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24618-3_18
11. Garg, A., Tamassia, R.: On the computational complexity of upward and rectilinear planarity testing. *SIAM J. Comput.* **31**(2), 601–625 (2002)
12. Harrigan, M., Healy, P.: Practical level planarity testing and layout with embedding constraints. In: Hong, S.-H., Nishizeki, T., Quan, W. (eds.) *GD 2007*. LNCS, vol. 4875, pp. 62–68. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-77537-9_9
13. Jünger, M., Leipert, S.: Level planar embedding in linear time. In: Kratochvíl, J. (ed.) *GD 1999*. LNCS, vol. 1731, pp. 72–81. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-46648-7_7
14. Klemz, B., Rote, G.: Ordered level planarity, geodesic planarity and bi-monotonicity. In: Frati, F., Ma, K.-L. (eds.) *GD 2017*. LNCS, vol. 10692, pp. 440–453. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73915-1_34
15. Leipert, S.: *Level Planarity Testing and Embedding in Linear Time*. Ph.D. thesis, University of Cologne (1998)

Schematizing Regional Landmarks for Route Maps

Marcelo Galvão^(✉), Jakub Krukar, and Angela Schwering

Institute for Geoinformatics, University of Münster, Münster, Germany
`galvao.marcelo@uni-muenster.de`

1 Introduction

Regional landmarks play an important role in facilitating wayfinding and orientation in navigation tasks [7]. Information such as “the route goes around the city center” or “there is a right turn after going along the park” represent spatial relations between polygonal objects and a route. Schematic visualizations are used for the cognitively adequate representation of such spatial relations [4]. Some commercial schematic maps make use of regional landmarks (urban areas, parks, lakes, forests). In contrast to topographic maps, they change shape, orientation and scale of the polygons to emphasize their topological function.

Although regional landmarks are used in commercial maps, related work focusing on route [1, 3] or network schematization [5, 6, 10] does not consider such landmarks at all. Publications addressing schematization of regional landmarks or subdivisions [2, 9] do not consider their spatial relation with paths. Since such landmarks are more important as references in route maps for drivers than in transit maps, there is a need for an algorithm that can produce route maps with regional landmarks, highlighting their correct spatial relation with the path.

In this contribution, we describe a new approach for drawing polygonal landmarks over an already schematized route. For the topological correct schematization, the method makes use of affine transformations and an adaptation of Nöllenburg-Wolff’s Mixed Integer Programming (MIP) for metro map drawing [6]. The advantage of using MIP over Buchin et al. method for polygon schematization [2] is that it allows results with higher level of abstraction by enforcing the correct topology with hard constraints while aesthetics are optimized in the objective function. We are able to emphasize crossings by constraining angles, and line alongness¹ by manipulating control points. The results resemble regional landmarks drawn by designers in commercial schematic maps.

2 Approach

To test the proposed method we read and planarize OpenStreetMap route data and polygonal geometries in its surrounding area, such as parks, lakes, urban

This research was supported by the ERC StRG Grant Agreement No 637645.

¹ Line alongness: the ratio of the region boundary being parallel to a path.

areas. For polygons overlapping the route, extra vertices are added at each crossing dividing the polygon into paths (sections). For polygons disconnected from the route, we select two vertices of the polygon as control points and two corresponding extra vertices are added to the route in order to hold their relative positions. The control points are selected based on their distance to the route (d), and the distance to the beginning/end of the linear referencing of the polygon against the route (l). We want d and l to be simultaneously minimized.

After the planarization process is completed the route is schematized. The schematization rescales the route parts, restricts edges orientation to the octilinear angles, forces fixed angles at decision points, while bends and shape distortion are minimized at the same time. Details on route schematization process are omitted for space reasons but we use similar approach as in [3].

With the newly calculated schematic position of the route vertices, we use affine transformation to transpose the polygons. For polygons crossed by the route, the transformation is made to their resulting paths independently, and the crossings themselves are used as references by the transformation. For polygons disconnected from the route, the pair of control points and their correspondent vertices in the route are used to readjust their position. We allow this adjustment to be looser or tighter depending on the spatial relation we want to highlight.

After the affine transformation is applied to the polygons or their subsections, the transposed geometries are submitted to the schematization process. The schematized geometry of the route is sent to the polygon schematization process to preserve their mutual topology. For the schematization process, we adapt Nöllenburg-Wolff's MIP inequalities. We use the hard constraints for octilinearity and edge spacing [6] to ensure the correct topology with the route. One limitation of the edge spacing constraint, is that it requires the route and the polygons to have the same edge orientation restriction.

For the objective function, we combine three soft constraints that are summed together and can be weighted by independent parameters. To enhance similarity to the original shape of the polygon, we use Nöllenburg-Wolff's function to preserve relative positions. Additionally, we add a new function that preserve location by minimizing distance between old and the new polygon vertices position in the L_1 -norm. To enhance abstraction, we use a similar bend cost function as Nöllenburg-Wolff's one, which penalizes bends along the resulting polygon shape. That way similarity and abstraction is balanced by adjusting the parameters.

3 Conclusion and Future Work

Using our application and real data of Münsterland-Germany, we were able to create drawings of schematized regional landmarks that emphasizes particular spatial relations with a route (e.g, line alongness and crossings). Next, we want to formalize the approach for the ten groups of path-polygon topological relations described by Shariff et al. [8], and later extend the method to be applicable with more complex street networks. Finally, we want to develop empirical experiments

with participants to test how the spatial relations are interpreted and recalled in navigation tasks as compared to topographic maps.

References

1. Agrawala, M., Stolte, C.: Rendering effective route maps: improving usability through generalization. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, vol. 1, pp. 241–249. ACM (2001). <https://doi.org/10.1145/383259.383286>
2. Buchin, K., Meulemans, W., Speckmann, B.: A new method for subdivision simplification with applications to urban-area generalization. In: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 261–220. ACM (2011). <https://doi.org/10.1145/2093973.2094009>
3. Delling, D., Gemsa, A., Nöllenburg, M., Pajor, T., Rutter, I.: On d-regular schematization of embedded paths. *Comput. Geom. Theory Appl.* **47**(3), 381–406 (2014)
4. Klippel, A., Richter, K.F., Barkowsky, T., Freksa, C.: The cognitive reality of schematic maps. In: Meng, L., Reichenbacher, T., Zipf, A. (eds.) *Map-based Mobile Services*, pp. 55–71. Springer, Heidelberg (2005). <https://doi.org/10.1007/3-540-26982-7-5>
5. Kopf, J., Agrawala, M., Bargerion, D., Salesin, D., Cohen, M.: Automatic generation of destination maps. *ACM Trans. Graph.* **29**(6), 1 (2010). <https://doi.org/10.1145/1882261.1866184>
6. Nöllenburg, M., Wolff, A.: Drawing and labeling high-quality metro maps by mixed-integer programming. *IEEE Trans. Vis. Comput. Graph.* **17**(5), 626–641 (2011). <https://doi.org/10.1109/TVCG.2010.81>
7. Schwering, A., Krukar, J., Li, R., Anacta, V.J., Fuest, S.: Wayfinding through orientation. *Spat. Cogn. Comput.* **17**(4), 273–303 (2017). <https://doi.org/10.1080/13875868.2017.1322597>
8. Shariff, A.R.B., Egenhofer, M.J., Mark, D.M.: Natural-language spatial relations between linear and areal objects: the topology and metric of English-language terms. *Int. J. Geograph. Inf. Sci.* **12**(3), 215–246 (1998). <https://doi.org/10.1.1.16.918>
9. Van Goethem, A., Meulemans, W., Speckmann, B., Wood, J.: Exploring curved schematization of territorial outlines. *IEEE Trans. Vis. Comput. Graph.* **21**(8), 889–902 (2015). <https://doi.org/10.1109/TVCG.2015.2401025>
10. Wang, Y.S., Chi, M.T.: Focus plus context metro maps. *IEEE Trans. Vis. Comput. Graph.* **17**(12), 2528–2535 (2011). <https://doi.org/10.1109/TVCG.2011.205>

Low-Degree Graphs Beyond Planarity

Patrizio Angelini^(✉), Michael A. Bekos, Michael Kaufmann,
and Thomas Schneck

Institut für Informatik, Universität Tübingen, Tübingen, Germany
{angelini,bekos,mk,schneck}@informatik.uni-tuebingen.de

Abstract. We study beyond-planarity for graphs of low degree. In particular, we aim at establishing tight bounds for values of d such that every graph of degree at most d belongs to a certain beyond-planarity class.

Beyond-planarity is a central topic in graph drawing, studying algorithmic and combinatorial properties of non-planar graphs. The most-studied beyond-planarity classes include: (i) *k-planar graphs*, where each edge crosses at most k edges, (ii) *quasiplanar graphs*, which disallow 3 mutually crossing edges, (iii) *fan-planar graphs*, where an edge only crosses a *fan* (a set of edges incident to a common vertex), (iv) *fan-crossing-free graphs*, where no edge crosses a fan, and (v) *RAC k-bend graphs*, where crossings happen at right angles and edges have at most k bends. For further definitions and state of the art, see [11].

Our goal is to establish upper and lower bounds for values of d such that every graph of degree at most d belongs to a certain beyond-planarity class. Table 1 summarizes the state of the art, including our results.

To prove that for any fixed $k > 0$ there exists an infinite family of bipartite Hamiltonian degree-3 graphs whose members are not k -planar, we employ an argument based on the crossing number of the n -vertex 3-regular graph known in the literature [14] as *cube-connected cycles* CCC_n . This graph is constructed starting from the n -regular *hypercube graph* [12] $Q_n = (V_n, E_n)$, whose 2^n vertices are denoted by distinct n -digit binary numbers; then, two vertices are joined

Table 1. The largest (second column) and smallest (third) value of d such that all (not all) degree- d graphs belong to certain beyond-planarity classes.

Graph class	Feasible	Infeasible
k -planar Hamiltonian bipartite	2	3 (CCC_n , Theorem 1)
fan-planar Hamiltonian bipartite	2	3 (CCC_n , Corollary 1)
quasi-planar	4 [2]	10 (K_{11} , ref. [1])
RAC (0-bend)	2	4 ($K_{4,4}$, ref. [10])
RAC (0-bend) Hamiltonian	3 [5]	4 ($K_{4,4}$, ref. [10])
RAC 1-bend	3 [4]	9 (K_{10} , ref. [3])
RAC 2-bends	6 [4]	148 (K_{149} , ref. [6])
fan-crossing-free	3 [2]	5 ($K_{5,5}$, Theorem 2)

by an edge in E_n if and only if their binary representations differ in a single digit. To obtain CCC_n , each vertex v of Q_n is replaced with a cycle of length n .

Graph CCC_n has $n2^n$ vertices and $3n2^{n-1}$ edges, and its crossing number is known [15] to be larger than $\frac{1}{20}4^n - (9n+1)2^{n-1}$. Hence, there is an edge with at least $\lceil \frac{1}{15} \frac{2^n}{n} - 6 - \frac{2}{2n} \rceil$ crossings, which shows that, for every $k \geq 1$, graph CCC_n is not k -planar for every n so that $k < \lceil \frac{1}{15} \frac{2^n}{n} - 6 - \frac{2}{2n} \rceil$. Since, for even values of $n \geq 6$, graph CCC_n is bipartite and Hamiltonian [13], we have the following.

Theorem 1. *For every $k \geq 1$, there exist infinitely many bipartite Hamiltonian 3-regular graphs that are not k -planar.*

Note that Theorem 1 could also be derived from random graph theory [8].

As observed in [2], every degree-4 graph is quasiplanar, since it has thickness 2. Thus, Theorem 1 provides an alternative proof that, for any fixed k , there exist quasiplanar graphs that are not k -planar [7]. Further, since every fan-planar drawing of a 3-regular graph is a 3-planar drawing, we have the following.

Corollary 1. *There exist infinitely many 3-regular bipartite Hamiltonian graphs that are not fan-planar.*

Alam et al. [2] observed that every degree-3 graph that can be decomposed into a matching and a set of cycles is fan-crossing-free and quasiplanar at the same time. This result can be extended to every degree-3 graph as follows¹. First, contract vertices of degree at most 2 and remove self-loops and bridges, to obtain a 3-regular bridgeless simple graph, which admits the required decomposition by Petersen’s theorem; then, reinsert the contracted or removed edges while maintaining the fan-crossing-free and quasi-planarity properties.

We prove that this result cannot be extended to degree-5 graphs, by showing that the 5-regular complete bipartite graph $K_{5,5}$ is not fan-crossing free. We prove this by means of a stronger result, namely a characterization of the complete bipartite fan-crossing-free graphs, analogous to existing characterizations for other beyond-planarity classes [9, 10].

Theorem 2. *The complete bipartite graph $K_{a,b}$, with $a \leq b$, is fan-crossing-free if and only if (i) $a \in \{1, 2\}$, or (ii) $a \in \{3, 4\}$ and $b \leq 6$. In particular, $K_{5,5}$ is not fan-crossing-free.*

We pose as future goal to further narrow the gaps between the bounds described in Table 1. In particular, the main open question is whether degree-3 graphs are RAC; this long-standing question has been posed already several times and is the one that first triggered our study. Note that the fan-crossing-free and quasiplanarity properties are necessary conditions for a graph to be RAC. In this sense, the extension of the result by Alam et al. [2] to all degree-3 graphs is an important step towards an answer to this question. Another intriguing question that stems from our results is whether degree-4 graphs are fan-crossing-free. Finally, the upper bounds for d concerning quasiplanar, RAC

¹ We thank an anonymous reviewer of GD’18 for suggesting this extension.

1-bend, and RAC 2-bend graphs presented in Table 1 descend from the known upper bounds on the maximum edge density of graphs in these classes [1, 3, 6]; it would be interesting to prove the existence of some low-degree graphs not belonging to these classes by exploiting direct arguments.

References

1. Ackerman, E., Tardos, G.: On the maximum number of edges in quasi-planar graphs. *J. Comb. Theory, Ser. A* **114**(3), 563–571 (2007). <https://doi.org/10.1016/j.jcta.2006.08.002>
2. Alam, M.J., et al.: Working group B2: beyond-planarity of graphs with bounded degree. In: Hong, S.-H., Kaufmann, M., Kobourov, S.G., Pach, J. (eds.) *Beyond-Planar Graphs: Algorithmics and Combinatorics (Dagstuhl Seminar 16452)*, Dagstuhl Reports, vol. 6, pp. 55–56. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2017)
3. Angelini, P., Bekos, M., Förster, H., Kaufmann, M.: On RAC drawings of graphs with one bend per edge. In: *International Symposium on Graph Drawing (GD 2018)* (2018)
4. Angelini, P., Cittadini, L., Didimo, W., Frati, F., Di Battista, G., Kaufmann, M., Symvonis, A.: On the perspectives opened by right angle crossing drawings. *J. Graph Algorithms Appl.* **15**(1), 53–78 (2011)
5. Argyriou, E.N., Bekos, M.A., Kaufmann, M., Symvonis, A.: Geometric RAC simultaneous drawings of graphs. *J. Graph Algorithms Appl.* **17**(1), 11–34 (2013). <https://doi.org/10.7155/jgaa.00282>
6. Arikushi, K., Fulek, R., Keszegh, B., Moric, F., Tóth, C.D.: Graphs that admit right angle crossing drawings. *Comput. Geom.* **45**(4), 169–177 (2012). <https://doi.org/10.1016/j.comgeo.2011.11.008>
7. Bae, S.W., et al.: Gap-planar graphs. *Theor. Comput. Sci.* (2018). <https://doi.org/10.1016/j.tcs.2018.05.029>
8. Bollobás, B.: The isoperimetric number of random regular graphs. *Eur. J. Comb.* **9**(3), 241–244 (1988). [https://doi.org/10.1016/S0195-6698\(88\)80014-3](https://doi.org/10.1016/S0195-6698(88)80014-3)
9. Czap, J., Hudák, D.: 1-planarity of complete multipartite graphs. *Discrete Appl. Math.* **160**(4–5), 505–512 (2012). <http://dx.doi.org/10.1016/j.dam.2011.11.014>
10. Didimo, W., Eades, P., Liotta, G.: A characterization of complete bipartite RAC graphs. *Inf. Process. Lett.* **110**(16), 687–691 (2010). <http://dx.doi.org/10.1016/j.ipl.2010.05.023>
11. Didimo, W., Liotta, G., Montecchiani, F.: A survey on graph drawing beyond planarity. CoRR, abs/1804.07257, (2018). <http://arxiv.org/abs/1804.07257>
12. Harary, F., Hayes, J.P., Wu, H.-J.: A survey of the theory of hypercube graphs. *Comput. Math. Appl.* **15**(4):277–289 (1988). [https://doi.org/10.1016/0898-1221\(88\)90213-1](https://doi.org/10.1016/0898-1221(88)90213-1)
13. Hsu, L., Ho, T., Ho, Y., Tsay, C.: Cycles in cube-connected cycles graphs. *Discrete Appl. Math.* **167**, 163–171 (2014). <http://dx.doi.org/10.1016/j.dam.2013.11.021>
14. Preparata, F.P., Vuillemin, J.: The cube-connected cycles: a versatile network for parallel computation. *Commun. ACM* **24**(5), 300–309 (1981). <http://dx.doi.org/10.1145/358645.358660>
15. Sýkora, O., Vrtó, I.: On crossing numbers of hypercubes and cube connected cycles. *BIT* **33**, 232–237 (1993)

Bounding the Tripartite-Circle Crossing Number of Complete Tripartite Graphs

Charles Camacho¹, Silvia Fernández-Merchant², Marija Jelic³, Rachel Kirsch⁴,
Linda Kleist⁵(✉), Elizabeth Bailey Matson⁶, and Jennifer White⁷

¹ Oregon State University, Corvallis, USA
camachoc@math.oregonstate.edu

² California State University, Los Angeles, USA
silvia.fernandez@csun.edu

³ University of Belgrade, Belgrade, Serbia
marijaj@matf.bg.ac.rs

⁴ London School of Economics, London, UK
r.kirsch1@lse.ac.uk

⁵ Technische Universität Berlin, Berlin, Germany
kleist@math.tu-berlin.de

⁶ Alfred University, Alfred, USA
matson@alfred.edu

⁷ Saint Vincent College, Latrobe, USA
jennifer.white@stvincent.edu

1 Introduction

The *crossing number* of a graph G , denoted by $\text{cr}(G)$, is the minimum number of edge-crossings over all drawings of G on the plane. To date, even the crossing numbers of complete and complete bipartite graphs are open. For the crossing number of the complete bipartite graph Zarankiewicz [6] showed that

$$\text{cr}(K_{m,n}) \leq \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{m}{2} \right\rfloor \left\lfloor \frac{m-1}{2} \right\rfloor,$$

and conjectured that equality holds. Harary and Hill [4] and independently Guy [3] conjectured that the crossing number of the complete graph K_n is

$$\text{cr}(K_n) = \frac{1}{4} \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{n-2}{2} \right\rfloor \left\lfloor \frac{n-3}{2} \right\rfloor =: H(n).$$

The construction of Harary and Hill is a so-called *cylindrical drawing*, in which the vertices lie on the circles of a cylinder, and edges of the graph cannot cross the circles. Towards the Zarankiewicz Conjecture, these drawings can be restricted to *bipartite cylindrical drawings*, in which each set of the vertex partition lies on its own circle. A *k-circle drawing* of a graph G is a drawing of G in the plane where the vertices are placed on k disjoint circles and the edges do not cross the circles. The *k-circle crossing number* of a graph G is the minimum number of crossings in a k -circle drawing of G . For the special case when G is a

k -partite graph, we can further require that each of the k vertex classes is placed on one of the k circles. The corresponding crossing number is called the k -partite-circle crossing number and is denoted by $cr_{\mathbb{K}}(G)$. Richter and Thomassen [5] showed that $cr_{\mathbb{2}}(K_{n,n}) = n\binom{n}{3}$. Ábrego, Fernández-Merchant, and Sparks [1] generalized this result for $m \leq n$ to

$$cr_{\mathbb{2}}(K_{n,m}) = \binom{n}{2} \binom{m}{2} + \sum_{0 \leq i < j \leq m-1} \left(\lfloor \frac{n}{m} j \rfloor - \lfloor \frac{n}{m} i \rfloor \right) \left(\lfloor \frac{n}{m} j \rfloor - \lfloor \frac{n}{m} i \rfloor - n \right).$$

2 Our Results

We investigate the tripartite-circle crossing number of the complete tripartite graph. Drawings that minimize the number of crossings are *good*, i.e., no edge crosses itself and any two edges share at most one point. We develop methods to count the number of crossings in good drawings and provide concrete drawings to obtain upper bounds.

Theorem 1. *For any integers $m, n,$ and $p,$*

$$\begin{aligned} & \sum_{\substack{\{x,y\} \in \binom{\{m,n,p\}}{2} \\ z \in \{m,n,p\} \setminus \{x,y\}}} \left(cr_{\mathbb{2}}(K_{x,y}) + xy \left\lfloor \frac{z}{2} \right\rfloor \left\lfloor \frac{z-1}{2} \right\rfloor \right) \leq cr_{\mathbb{3}}(K_{m,n,p}) \\ & \leq \sum_{\substack{\{x,y\} \in \binom{\{m,n,p\}}{2} \\ z \in \{m,n,p\} \setminus \{x,y\}}} \left(\binom{x}{2} \binom{y}{2} + xy \left\lfloor \frac{z}{2} \right\rfloor \left\lfloor \frac{z-1}{2} \right\rfloor \right). \end{aligned}$$

For the balanced case, Fig. 1 illustrates the drawing, and the formulas simplify to

$$3n \binom{n}{3} + 3n^2 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \leq cr_{\mathbb{3}}(K_{n,n,n}) \leq 3 \binom{n}{2}^2 + 3n^2 \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor.$$

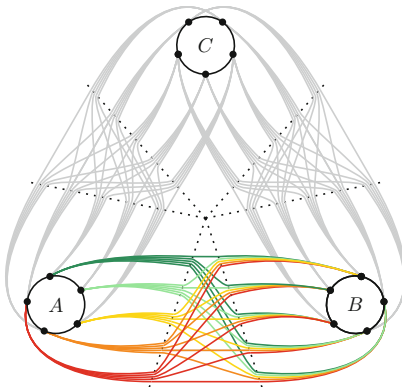


Fig. 1. A tripartite-circle drawing of $K_{n,n,n}$ proving the upper bound.

Connection to the Harary-Hill Conjecture The drawings of K_n presented by Harary and Hill [4] have $H(n)$ crossings and consist of a 2-circle drawing of $K_{n/2, n/2}$ together with all straight line segments joining vertices on the same circle. Moreover, Blažek and Koman [2] presented a 1-circle drawing of K_n with $H(n)$ crossings. Therefore it has been asked whether a 3-circle drawing of $K_{\frac{n}{3}, \frac{n}{3}, \frac{n}{3}}$ together with all straight line segments joining vertices on the same circle can achieve $H(n)$ crossings. Our result proves that such a drawing does not exist.

References

1. Ábrego, B.M., Fernández-Merchant, S., Sparks, A.: The cylindrical crossing number of the complete bipartite graph (2017, preprint)
2. Blažek, J., Koman, M.: A minimal problem concerning complete plane graphs. *Theory Graphs Appl. Czech. Acad. Sci.* 113–117 (1964)
3. Guy, R.K.: A combinatorial problem. *Nabla (Bull. Malayan Math. Soc.)* **7**, 68–72 (1960)
4. Harary, F., Hill, H.: On the number of crossings in a complete graph. *Proc. Edinburgh Math. Soc.* **13**, 333–338 (1963)
5. Richter, R.B., Thomassen, C.: Relations between crossing numbers of complete and complete bipartite graphs. *Am. Math. Mon.* **104–2**, 131–137 (1997)
6. Zarankiewicz, K.: On a problem of P. Turan concerning graphs. *Fundamenta Mathematicae* **41–1**, 137–145 (1955)

On the Edge Density of k -Planar Graphs

Steven Chaplick, Andre Löffler, and Rainer Schmöger^(✉)

Lehrstuhl für Informatik I, Universität Würzburg, Würzburg, Germany

rainer.schmoeger@stud-mail.uni-wuerzburg.de,

<http://www1.informatik.uni-wuerzburg.de/en/staff>

In the 18th century Euler discovered his famous polyhedron formula, which can be used to bound the edge density for planar graphs. Let $G = (V, E)$ be a simple and planar graph with $|V| \geq 3$, then $|E| \leq 3|V| - 6$. Turán co-established Extremal Graph Theory, a branch in which extremal graphs are investigated under the assumption of specified properties. He studied the edge density of graphs which are not necessarily planar but do not contain cliques of fixed size.

For several beyond-planar graph classes Turán-type results were discovered:

- k -planar graphs, for which there exists a drawing where no edge is crossed more than k times are studied in [2, 5, 10–12, 15, 16].
- k -quasi-planar graphs with no set of k pairwise crossing edges are investigated in [1, 3–5, 12, 13, 18, 19].
- Fan-planar graphs, where edges can be crossed by one fan, a set of edges sharing one common endpoint [5, 6, 8, 9, 14].

We consider the edge density of (non-) simple k -planar graphs. A *simple* graph does not contain loops or parallel edges. A *non-simple* multigraph has a drawing without homotopic parallel edges and self-loops. Bodendiek et al. first bounded the edge density of 1-planar graphs [10]. Pach and Tóth [16] gave bounds for k -planar graphs with $0 \leq k \leq 4$, namely $|E| \leq (k+3)(|V|-2)$, including Euler's result for planar graphs. Edge density of k -planar graphs strongly relates to the *Crossing Lemma* which provides a lower bound on the crossing number $cr(G)$ for any graph G . They [16] used their bounds to improve it to $cr(G) \geq \frac{1}{33.75} \cdot \frac{|E|^3}{|V|^2}$. Later Pach et al. [15] improved the bound for 3-planar graphs to $|E| \leq 5.5(|V|-2)$. A charging argument by Ackerman [2] improves the bound for 4-planar graphs to $|E| \leq 6(|V|-2)$, proving the current best constant ($\frac{1}{29}$) for the Crossing Lemma. For $k \geq 5$ only a general bound has been established: considering the number of crossings C , a lower bound by the Crossing Lemma and an upper bound from k -planarity yields (1) and gives $|E| \leq 3.807\sqrt{k}|V|$.

$$\frac{1}{29} \frac{|E|^3}{|V|^2} \leq cr(G) \leq C \leq \frac{|E| \cdot k}{2} \quad (1)$$

Pach et al. [17] recently proved a Crossing Lemma for multigraphs using another constant ($\approx 10^{-7}$), so a similar inequality to (1) leads to the following bound on the edge density of non-simple k -planar graphs. Curiously, this appears to be the first and only upper bound known for arbitrary k .

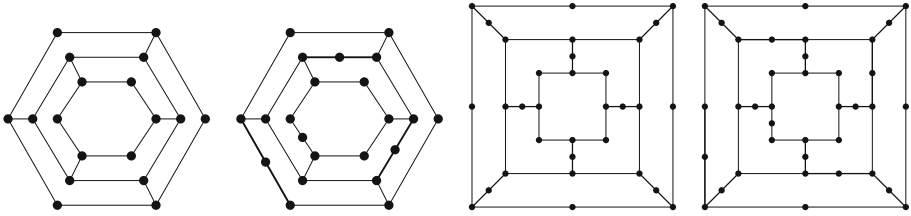


Fig. 1. True planar skeletons for different values of k (left-to-right): $k = 4$ [15]; $5 \leq k \leq 6$; $7 \leq k \leq 9$; $10 \leq k \leq 12$.

Table 1. Bounds on the number of edges for non-simple and simple graphs; similar bounds can also be obtained for larger values of k .

	Simple	[2, 16]	Non-simple	[Theorem 1]
k	Lower bound	Upper bound	Lower bound	Upper bound
5	$6.00 V - 16.00$	$8.51 V $	$6.20 V - 12.40$	$5000 V $
6	$6.80 V - 23.60$	$9.32 V $	$7.00 V - 14.00$	$5478 V $
7	$7.00 V - 20.00$	$10.07 V $	$7.33 V - 14.67$	$5917 V $
8	$7.33 V - 20.67$	$10.77 V $	$7.67 V - 15.33$	$6325 V $
9	$7.67 V - 21.33$	$11.42 V $	$8.00 V - 16.00$	$6709 V $
10	$7.71 V - 23.43$	$12.04 V $	$8.14 V - 16.29$	$7072 V $
11	$8.00 V - 24.00$	$12.63 V $	$8.43 V - 16.86$	$7417 V $
12	$8.57 V - 25.14$	$13.19 V $	$9.00 V - 18.00$	$7746 V $

Theorem 1. For $k \geq 1$, a non-simple k -planar graph G has $|E| < 2237\sqrt{k}|V|$.

We also construct lower bound examples based on the structure of *optimal* k -planar graphs ($k \leq 3$), i.e., k -planar graphs with maximum edge density. Namely, Bekos et al. [7] showed that every optimal non-simple 2-planar (3-planar) graph has a regular *true planar skeleton*: a spanning subgraph consisting of a set of crossing-free edges with only pentagonal (hexagonal) faces. In the original graph, every such face is (almost) a clique, having five (eight) edges inside.

The idea of the true planar skeleton leads us to lower bounds on edge density. Using the patterns in Fig. 1 for $4 \leq k \leq 12$ and adding all possible edges in every face respecting k -planarity produces a family of non-simple k -planar graphs, i.e., establishing lower bounds on the edge density. To use these skeletons for lower bounds on simple k -planar graphs, we have to carefully consider how to avoid multi edges when inserting edges. In particular, by using skeletons such as those shown in Fig. 1, we obtain new bounds for both simple and non-simple k -planar graphs – all discovered bounds can be found in Table 1.

Observe that the general upper bounds on the edge density of k -planar graphs rely on a naive upper bound on the number of crossings; see (1). Additionally,

our current best lower bound examples are based on true planar skeletons. These remarks lead to the following questions:

Q1 *Is there a better bound on the crossing number for optimal k -planar graphs?*

Q2 *Can we obtain better lower bounds if we do not use true planar skeletons?*

References

1. Ackerman, E.: On the maximum number of edges in topological graphs with no four pairwise crossing edges. In: Symposium on Computational Geometry, pp. 259–263. ACM (2006)
2. Ackerman, E.: On topological graphs with at most four crossings per edge. CoRR abs/1509.01932 (2015)
3. Ackerman, E., Tardos, G.: On the maximum number of edges in quasi-planar graphs. *J. Comb. Theory, Ser. A* **114**(3), 563–571 (2007)
4. Agarwal, P.K., Aronov, B., Pach, J., Pollack, R., Sharir, M.: Quasi-planar graphs have a linear number of edges. *Combinatorica* **17**(1), 1–9 (1997)
5. Angelini, P., Bekos, M.A., Kaufmann, M., Pfister, M., Ueckerdt, T.: Beyond-planarity: Density results for bipartite graphs. CoRR abs/1712.09855 (2017)
6. Bekos, M.A., Cornelsen, S., Grilli, L., Hong, S., Kaufmann, M.: On the recognition of fan-planar and maximal outer-fan-planar graphs. *Algorithmica* **79**(2), 401–427 (2017)
7. Bekos, M.A., Kaufmann, M., Raftopoulou, C.N.: On optimal 2- and 3-planar graphs. In: Symposium on Computational Geometry, LIPIcs, vol. 77, pp. 16:1–16:16. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2017)
8. Binucci, C., et al.: Algorithms and characterizations for 2-layer fan-planarity: from caterpillar to stegosaurus. *J. Graph Algorithms Appl.* **21**(1), 81–102 (2017)
9. Binucci, C., et al.: Fan-planarity: properties and complexity. *Theor. Comput. Sci.* **589**, 76–86 (2015)
10. Bodendiek, R., Schumacher, H., Wagner, K.: Über 1-optimale graphen. *Mathematische Nachrichten* **117**(1), 323–339 (1984)
11. Brandenburg, F.J., Eppstein, D., Gleißner, A., Goodrich, M.T., Hanauer, K., Reislhuber, J.: On the density of maximal 1-planar graphs. In: Didimo, W., Patrignani, M. (eds.) GD 2012. LNCS, vol. 7704, pp. 327–338. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36763-2_29
12. Chaplick, S., Kryven, M., Liotta, G., Löffler, A., Wolff, A.: Beyond outerplanarity. In: Frati, F., Ma, K.-L. (eds.) GD 2017. LNCS, vol. 10692, pp. 546–559. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73915-1_42
13. Fox, J., Pach, J., Suk, A.: The number of edges in k -quasi-planar graphs. *SIAM J. Discrete Math.* **27**(1), 550–561 (2013)
14. Kaufmann, M., Ueckerdt, T.: The density of fan-planar graphs. CoRR abs/1403.6184 (2014)
15. Pach, J., Radoicic, R., Tardos, G., Tóth, G.: Improving the crossing lemma by finding more crossings in sparse graphs. *Discrete Comput. Geom.* **36**(4), 527–552 (2006)
16. Pach, J., Tóth, G.: Graphs drawn with few crossings per edge. *Combinatorica* **17**(3), 427–439 (1997)
17. Pach, J., Tóth, G.: A crossing lemma for multigraphs. In: Symposium on Computational Geometry, LIPIcs, vol. 99, pp. 65:1–65:13. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2018)

18. Suk, A.: k -quasi-planar graphs. In: van Kreveld, M., Speckmann, B. (eds.) GD 2011. LNCS, vol. 7034, pp. 266–277. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-25878-7_26
19. Suk, A., Walczak, B.: New bounds on the maximum number of edges in k -quasi-planar graphs. *Comput. Geom.* **50**, 24–33 (2015)

Confluent* Drawings by Hierarchical Clustering

Jonathan X. Zheng^(✉), Samraat Pawar, and Dan F. M. Goodman

Imperial College London, London, UK
jxz12@ic.ac.uk

Recently an edge bundling technique known as confluent* drawing was applied to general graphs by Bach et al. [2] by leveraging power graph decomposition (a form of edge compression that groups similar vertices together, merging edges shared among group members). We explore the technique further by demonstrating the equivalence between confluent drawing and the hierarchical edge bundling of Holten [3], thereby opening the door for existing hierarchical clustering algorithms to be used instead of power graphs to produce confluent drawings for general graphs. We investigate various popular hierarchical clustering methods, and present a qualitative experimental comparison between them. We also introduce a new distance measure for agglomerative clustering that outperforms previous measures, and make recommendations for using the method in practice.

Creating the Routing Graph. The method of bundling we consider consists of two steps: first, we find a suitable auxiliary routing graph; second, we find a layout for this new graph, and then draw the original edges back on top, using invisible routing nodes as spline control points. An example of this can be seen in Fig. 1.

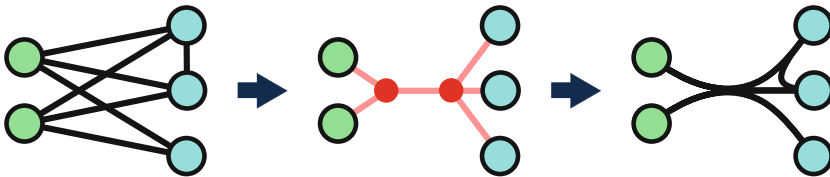


Fig. 1. An example of a simple graph (left), its potential routing graph (middle), and the resultant drawing with edges bundled through routing nodes (right).

This should not be confused with the routing graph used by Pupyrev et al. [4] in their paper on *metro-style* bundling, which generates a routing based on curving around fixed node positions. Our routing is generated using the topology of the graph itself. The benefit of this is that the bundling reflects the actual

The original definition of *confluent* by Dickerson et al. [1] forbids edge crossings, while Bach et al. [2] recognize but do not strictly follow this. We continue the use of such terminology here for consistency, but recognize its imprecise usage. A more suitable name for the general edge bundling method of using an auxiliary routing graph should be adopted in the future.

structure of the data, rather than a potentially arbitrary spatial positioning. On the other hand, the popular method of Holten [3] generates its routing using a hierarchy already included in the data, which is easily converted into a graph through a tree representation, where original vertices are the leaves and groups within the hierarchy are branch nodes.

The primary purpose of this poster is to show that the work on confluent drawings by Bach et al. [2] is also based on a routing graph, and therefore should be classified with these previous two techniques, rather than within the realm of confluent drawing. As such, the routing graphs generated here may also be used to produce metro-style bundles [4].

As noted by Bach et al. [2] in their original paper, we also find that power graph decomposition performs poorly on graphs where clusters or cliques are a common motif, producing fractal-like artifacts (see poster for example). To alleviate this issue, we investigate the use of hierarchical clustering to generate the routing instead. This requires the definition of a dissimilarity measure between pairs of vertices, a popular choice being the Jaccard distance measure

$$d_{ij} = 1 - \frac{|N(i) \cap N(j)|}{|N(i) \cup N(j)|} \quad (1)$$

where $N(i)$ is the set of neighbours of vertex i . However, simply using Jaccard distance only captures the dissimilarity of vertices with shared neighbours, and any pair of vertices more than two hops away is automatically given a distance $d_{ij} = 1$. We introduce a method of capturing such longer range dissimilarities, by simply multiplying d_{ij} by the shortest path between them. A visual comparison between the two can be seen in the poster, along with a further example using a popular divisive clustering method.




Drawing the Bundled Graph. The result of a hierarchical clustering algorithm is a dendrogram (a rooted tree used to describe hierarchical relationships) which needs to be converted to a routing graph. One could simply assign unit edge lengths to the branches, but the output of agglomerative methods also includes a merging cost between clusters. We encode this using varying edge lengths, and therefore require a force-directed method that explicitly includes this. In our case we use a multidimensional scaling approach i.e. the popular Kamada-Kawai layout. This was recently improved by Zheng et al. [5] by making use of stochastic gradient descent, and can also be used to easily produce optimal radial layouts (see poster for example).

The original edges are then drawn on top of this layout using b-splines controlled along the shortest path through the routing graph. To improve the drawing aesthetically, we also reduce the bundling strength as in Holten [3], while full strength bundling can be used to reproduce the confluent effect utilised by Bach et al. [2].

References

1. Dickerson, M., Eppstein, D., Goodrich, M.T., Meng, J.Y.: Confluent drawings: visualizing non-planar diagrams in a planar way. *J. Graph Algorithms Appl.* **9**(1), 31–52 (2005). <https://doi.org/10.7155/jgaa.00099>
2. Bach, B., Riche, N.H., Hurter, C., Marriott, K., Dwyer, T.: Towards unambiguous edge bundling: investigating confluent drawings for network visualization. *IEEE Trans. Vis. Comput. Graph.* **23**(1), 541–550 (2017). <https://doi.org/10.1109/TVCG.2016.2598958>
3. Holten, D.: Hierarchical edge bundles: visualization of adjacency relations in hierarchical data. *IEEE Trans. Vis. Comput. Graph.* **12**(5), 741–748 (2006). <https://doi.org/10.1109/TVCG.2006.147>
4. Pupyrev, S., Nachmanson, L., Bereg, S., Holroyd, A.E.: Edge routing with ordered bundles. *Comput. Geom. Theory Appl.* **52**, 18–33 (2016). <https://doi.org/10.1016/j.comgeo.2015.10.005>
5. Zheng, X.J., Pawar, S., Goodman, D.F.M.: Graph drawing by stochastic gradient descent. *IEEE Trans. Vis. Comput. Graph.* (2018, to appear). <https://doi.org/10.1109/TVCG.2018.2859997>

Examining Weak Line Covers with Two Lines in the Plane

Oksana Firman, Fabian Lipp , Laura Straube , and Alexander Wolff 

Lehrstuhl für Informatik I, Universität Würzburg, Würzburg, Germany
{oksana.firman,fabian.lipp}@uni-wuerzburg.de, laura.straube@gmx.net

Chaplick et al. [2] defined the l -dimensional affine line cover number $\rho_d^l(G)$, for $1 \leq l < d$ and an arbitrary graph G , as the minimum number of l -dimensional planes in \mathbb{R}^d such that G admits a crossing-free straight-line drawing whose vertices and edges are contained in the union of these planes. The l -dimensional weak affine line cover number $\pi_d^l(G)$ also counts such planes but insists only that the vertices are covered by their union. In particular, the weak line cover number $\pi_2^1(G)$ is the minimum number of lines in the plane that are necessary to cover the vertices of a planar graph G .

Firman et al. [3] asked whether π_2^1 has a sublinear upper bound for the class of planar graphs. In the following we restrict their open problem further and make some progress in characterizing the class of graphs that can be drawn on two lines in the plane (further referred to as *drawable*). In order to verify conjectures (such as Conjecture 1 below), we needed a drawability test. Given that drawability is NP-hard to decide [1], we contented ourselves with exponential-time approaches.

First, we formulated drawability as an integer linear program (ILP). The solution of the ILP yields a drawing on two lines. Without loss of generality, we consider the case that the two lines are perpendicular and view their intersection point as the origin of a Euclidean coordinate system with four quadrants each incident to two half-axes. There are Boolean variables for every combination of a vertex and a half-axis describing whether they are incident. Other variables represent the order of the vertices on a given half-axis. The constraints ensure that every vertex is mapped to exactly one half-axis, that the ordering on each half-axis is transitive, and that the resulting drawing is planar.

Second, we transformed our ILP formulation into a Boolean formula in CNF that can be tested by a SAT solver. The ILP formulation uses only binary variables and was therefore easy to transform. Our hope was that the SAT formulation could be evaluated more efficiently than the ILP formulation. On our test suite with 824 solvable and 304 unsolvable graphs, the SAT solver MiniSat (version 2.2.0) was indeed always faster than the ILP solver IBM ILOG CPLEX Optimization Studio (12.8.0.0). Both in terms of total computation time and only solving time, the difference in speed was an order of magnitude.

Our experiments suggest the following.

Conjecture 1. Every planar graph with maximum degree 3 is drawable.

However, David Eppstein found a 3-regular counterexample with 26 vertices; see Fig. 3. We verified it using our SAT formulation.

We can show that any graph that contains nested triangles in each of its planar embeddings is not drawable. For graphs with maximum degree 3, this case does not arise as they can always be embedded such that there are no nested triangles due to their low connectivity, see Fig. 1. On the other hand, triangulations are not drawable – apart from the tetrahedron and graphs that extend the tetrahedron in a specific way; see Fig. 2. The graph with the dashed edges in Fig. 1 shows that not all 2-outerplanar and not all graphs of maximum degree 4 are drawable. We also managed to extend our nested-triangle condition to nested cycles; this yielded quadrangulations (which are obviously triangle-free) that are not drawable.

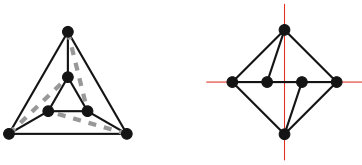


Fig. 1. The triangular prism (solid edges, left) is drawable (right), but its 4-regular supergraph with the dashed edges is non-drawable.

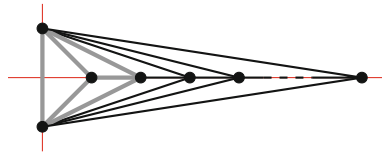


Fig. 2. The tetrahedron (in gray) and a family of drawable triangulations based on it.

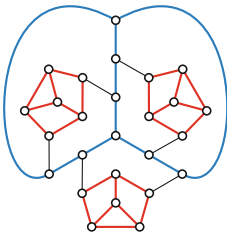


Fig. 3. David Eppstein's 3-regular graph that is not drawable on two lines.

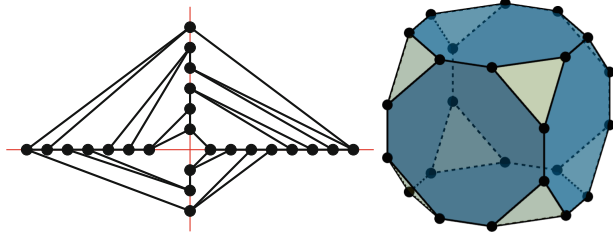


Fig. 4. The truncated hexahedron (a 3-regular graph with 24 vertices) drawn on two lines not using their intersection – and its representation as an Archimedean solid.

Our experiments also showed that all tested graphs of maximum degree 3 except the tetrahedron are not only drawable, but can be drawn on two lines such that no edge contains the intersection of the two lines; see, for example, Fig. 4. This could help in finding a strategy for distributing vertices on two lines. Under this additional condition no triangulation we tested was drawable, including any of those in Fig. 2.

References

1. Biedl, T., et al.: Line and plane cover numbers revisited (2018). Unpublished manuscript
2. Chaplick, S., Fleszar, K., Lipp, F., Ravsky, A., Verbitsky, O., Wolff, A.: Drawing graphs on few lines and few planes. In: Hu, Y., Nöllenburg, M. (eds.) GD 2016. LNCS, vol. 9801, pp. 166–180. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-50106-2_14
3. Firman, O., Ravsky, A., Wolff, A.: On the weak line cover numbers. In: Korman, M., Mulzer, W. (eds.) Proceedings of 34th European Workshop on Computational Geometry (EuroCG 2018), pp. 63:1–63:5 (2018). https://conference.imp.fu-berlin.de/eurocg18/download/paper_63.pdf

1-Gap Planarity of Complete Bipartite Graphs

Christian Bachmaier, Ignaz Rutter, and Peter Stumpf^(✉)

Faculty of Computer Science and Mathematics,
University of Passau, Passau, Germany
{bachmaier,rutter,stumpf}@fim.uni-passau.de

Abstract. A graph is 1-gap planar if it admits a drawing such that each crossing can be assigned to one of the two involved edges in such a way that each edge is assigned at most one crossing. We show that $K_{3,14}$, $K_{4,10}$ and $K_{6,6}$ are not 1-gap planar.

1 Introduction

A graph is 1-gap planar if it admits a drawing such that each crossing can be assigned to one of the two involved edges in such a way that each edge is assigned at most one crossing. The motivation comes from edge casings, where one creates a small gap in one of the edges involved in each crossing to increase the readability. In a 1-gap planar drawing each edge receives at most one such gap. This notion was introduced in GD'17 by Bae et al. [1]. Among others they showed that a 1-gap planar graph on n vertices has at most $5n - 10$ edges and this is tight. They further show that the complete graph K_n is 1-gap planar if and only if $n \leq 8$. An important observation of Bae et al. is that every 1-gap planar graph G satisfies $\text{cr}(G) \leq |E|$ (since each crossing is assigned to one of the edges). For complete bipartite graphs, they gave 1-gap planar drawings for $K_{3,12}$, $K_{4,8}$ and $K_{5,6}$, whereas they exclude $K_{3,15}$, $K_{4,11}$ and $K_{6,7}$ by observing that their crossing number is strictly greater than their edge number. They leave the remaining complete bipartite graphs as an open problem. We show the following theorem.

Theorem 1. *The graphs $K_{3,14}$, $K_{4,10}$ and $K_{6,6}$ are not 1-gap planar.*

This shrinks the open cases to $K_{3,13}$ and $K_{4,9}$. We note that for all the graphs we exclude, the crossing number equals the edge number [2]. Thus, we know that in a 1-gap planar drawing of such a graph each edge has at least one crossing.

2 Proof Strategy

Our proof strategy is an extension of the one of Bae et al., who encountered a similar situation when treating the case of K_9 , which has 36 edges and whose crossing number is 36. For convenience, we briefly sketch their argument. Assume for the sake of contradiction that Γ is a 1-gap planar drawing of K_9 , and consider

the planarization Γ^* of this drawing, where all crossings are replaced by dummy vertices. Observe that Γ has precisely $\text{cr}(K_9) = |E(K_9)| = 36$ crossings [1]. If two vertices of K_9 share a face in Γ^* , we can reroute the edge between them without crossings in this face, thus obtaining a drawing with fewer crossings, which is not possible. Thus, for any two original vertices their incident faces of Γ^* are disjoint. This gives a lower bound of 72 faces. On the other hand, from Euler’s formula it follows that Γ^* has only 65 faces; a contradiction.

In contrast, for complete bipartite graphs, vertices may share a face of the planarization if they are independent. Let $G = (R \cup B, E)$ be a complete bipartite graph with $\text{cr}(G) = |E| = |R| \cdot |B|$. The vertices in R and B are *red* and *blue*, respectively. As before, we consider a hypothetical 1-gap planar drawing Γ of G , for which we know that it has $\text{cr}(\Gamma) = \text{cr}(G) = |E|$ crossings, and we denote the planarization by Γ^* . Let F denote the set of faces of Γ^* and let $F_R, F_B \subseteq F$ be the faces that are incident to a red and a blue vertex, respectively. If $F_R \cap F_B \neq \emptyset$, then there is a face in F that is incident to both a red and a blue vertex. We can route the edge between them without crossings and thus reach a contradiction as in the case of K_9 . By assumption, Γ^* has $|R| + |B| + |E|$ vertices and $|R| \cdot |B| + 2 \cdot |E|$ edges, and hence $|F| = 2 \cdot |R| \cdot |B| - |R| - |B| + 2$ faces.

Consider the auxiliary bipartite graph $G_R = (R \cup F_R, E_R)$ where a face and a vertex are adjacent if and only if they are incident in Γ^* . The graph $G_B = (B \cup F_B, E_B)$ is defined analogously. Observe that $|E_R| = |E_B| = |R| \cdot |B|$ since each vertex in R has degree $|B|$ and vice versa. We argue that either G_R and G_B are both trees, or one of them, say G_R , is a cycle decorated with leaves in F_R and the other one, G_B , is a forest with two connected components.

In the former case, we obtain $|E_R| = |R| + |F_R| - 1$, which gives $|F_R| = |R| \cdot |B| - |R| + 1$ and likewise $|F_B| = |R| \cdot |B| - |B| + 1$. Hence $|F_B| + |F_R| = 2 \cdot |B| \cdot |R| - |R| - |B| + 2 = |F|$. In the latter case, the number of faces in F_R decreases by 1, but the number of faces in F_B increases by 1. In all cases we find that $|F_R| + |F_B| = |F|$, i.e., each face of Γ^* is either in F_R or in F_B . A contradiction is reached by showing that there exists at least one *white face* of Γ^* that is not incident to any red or blue vertex.

First it follows from the fact that each edge has a gap that there is a cycle C in Γ^* that only contains dummy vertices. This can be seen as follows. We start in any dummy vertex and follow the edge that does not have its gap there to its own gap. Repeating this step eventually produces the desired cycle C . If all red and blue vertices lie inside (outside) C , then C contains a white face in its exterior (interior). Otherwise it separates a component of G_R from a component of G_B . Further analysis yields a contradiction. The details vary depending on whether G is $K_{3,14}$, $K_{4,10}$ or $K_{6,6}$ as well as on the size and structure of the components that are separated by C .

3 Conclusion

We have shown that $K_{3,14}$, $K_{4,10}$ and $K_{6,6}$ are not 1-gap planar. We leave open the cases of $K_{3,13}$ and $K_{4,9}$. It seems difficult to adapt our proof technique

to these cases since their crossing numbers are strictly smaller than their edge number, which results in additional freedom for possible 1-gap planar drawings.

References

1. Bae, S.W., et al.: Gap-planar graphs. *Theor. Comput. Sci.* (2018). <https://doi.org/10.1016/j.tcs.2018.05.029>
2. Kleitman, D.J.: The crossing number of $K_{5,n}$. *J. Comb. Theory* **9**(4), 315–323 (1970). [https://doi.org/10.1016/S0021-9800\(70\)80087-4](https://doi.org/10.1016/S0021-9800(70)80087-4)

Extending Drawings of K_n into Arrangements of Pseudocircles

Alan Arroyo^{1,3(✉)}, R. Bruce Richter¹, and Matthew Sunohara²

¹ Department of Combinatorics and Optimization,
University of Waterloo, Waterloo, Canada
{amarroyo,brichter}@uwaterloo.ca

² Department of Mathematics, University of Toronto, Toronto, Canada
matthew.sunohara@gmail.com

³ IST Austria, Klosterneuburg, Austria

The Harary-Hill Conjecture states that the crossing number of the complete graph K_n is equal to:

$$H(n) = \frac{1}{4} \left\lfloor \frac{n}{2} \right\rfloor \left\lfloor \frac{n-1}{2} \right\rfloor \left\lfloor \frac{n-2}{2} \right\rfloor \left\lfloor \frac{n-3}{2} \right\rfloor.$$

In general, if \mathcal{S} is the unit sphere in \mathbb{R}^3 , then a *spherical drawing* of a graph G is one in which the vertices of G are represented as distinct points in \mathcal{S} , and every edge is a shortest-arc connecting its corresponding ends. Although the Harary-Hill Conjecture is known to be true for certain classes of drawings of K_n , it is yet unknown that spherical drawings have at least $H(n)$ crossings.

In the proofs of [1,3] showing that rectilinear drawings of K_n have at least $H(n)$ crossings, a crucial point was to relate the number of crossings in a given drawing to the separation properties of the $\binom{n}{2}$ lines extending the edges. Understanding these separation properties, but for the curves extending the edges in spherical drawings, serve as our motivation for studying arrangement of pseudocircles extending the edges of a drawing.

An *arrangement of pseudocircles* is a set of simple closed curves in the sphere in which every two curves intersect at most twice, and every intersection is a crossing. If γ is a simple closed curve, then a *side* of γ is one of the two disks in \mathcal{S} bounded by γ . In spherical drawings, the great circles extending the edge-arcs form an arrangement of pseudocircles.

With the aim of finding a combinatorial extension of spherical drawings analogous to how pseudolinear drawings extend rectilinear drawings, there have been two significant questions under active consideration:

- (Q1) Do the edges of every good drawing of K_n in the sphere extend to an arrangement of pseudocircles?
- (Q2) If the edges of a drawing of K_n extend to an arrangement of pseudocircles, is there an extending arrangement in which any two pseudocircles intersect exactly twice?

These questions were indeed considered by a working group at the 2015 Crossing Number Workshop in Rio de Janeiro.

In this work, we answer these questions by showing that (1) there is a drawing of K_{10} (Fig. 1c) in which there is no extension of its edges into an arrangement of pseudocircles; and (2) there is a drawing of K_9 in which there is an extension of its edges into an arrangement of pseudocircles, but no such extension exists for which any two curves cross exactly twice.

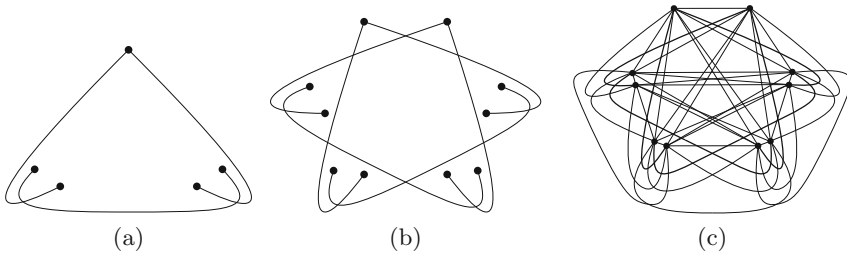


Fig. 1. Construction of a K_{10} with whose edges cannot be extended to an arrangement of pseudocircles.

To construct the examples we consider the *basic gadget* in Fig. 1a. This has the property that if we extend its three edges into an arrangement of pseudocircles, then either the pseudocircle extending the edge with two degree 1 vertices is drawn in the bounded face of the drawing or the two pseudocircles extending the other two edges are drawn in the bounded face. Overlapping two basic gadgets as in Fig. 1b yields a drawing not extendible to an arrangement of pseudocircles; Fig. 1b can be enlarged to the non-extendible drawing of K_{10} in Fig. 1c. A similar construction, but using two disjoint copies of the basic gadget, yields the drawing of K_9 answering (Q2) in the negative form.

Among the five non-isomorphic drawings of K_5 in the sphere, there are two that are non-rectilinear. The class of *convex drawings* of K_n is obtained by forbidding the two non-rectilinear K_5 s. Convex drawings (or locally rectilinear drawings) were introduced in [2] in the context of the Harary-Hill Conjecture, where it is shown that there is a possibility that every optimal drawing of K_n is convex.

In this work we show that every h-convex drawing, a special kind of convex drawing, can be extended into an arrangement of pseudocircles. Furthermore, the extension satisfies that if two vertices x and y are on the same side of a curve extending an edge, then the edge xy is drawn on that side of the curve. Moreover, we prove that any drawing of K_n with a pseudocircular extension of such kind is h-convex.

Related to (Q2), we also show that h-convex drawings have a “better” extension in which pseudocircles are pairwise intersecting. This, and the fact that h-convex drawings can be decomposed into two pseudolinear drawings, suggest that h-convex is, possibly, the right definition for pseudospherical drawings of K_n .

References

1. Ábrego, B.M., Fernández-Merchant, S.: A lower bound for the rectilinear crossing number. *Graphs Comb.* **21**, 293–300 (2005)
2. Arroyo, A., McQuillan, D., Richter, R.B., Salazar, G.: Convex drawings of K_n : topology meets geometry (submitted)
3. Lovász, L., Vesztergombi, K., Wagner, U., Welzl, E.: Convex quadrilaterals and k -sets. *Contemp. Math.* **342**, 139–148 (2004). Towards a theory of geometric graphs. Amer. Math. Soc., Providence, RI

Topology Based Scalable Graph Kernels

Kin Sum Liu^{1(✉)}, Chien-Chun Ni², Yu-Yao Lin³, and Jie Gao¹

¹ Stony Brook University, Stony Brook, NY 11794, USA
{kiliu, jgao}@cs.stonybrook.edu

² Yahoo Research, Sunnyvale, CA 94089, USA
cni02@oath.com

³ Intel Inc., Hillsboro, OR 97124, USA
yu-yao.lin@intel.com

1 Introduction

We propose a new graph kernel for graph classification and comparison using Ollivier Ricci curvature. The Ricci curvature of an edge in a graph describes the connectivity in the local neighborhood. An edge in a densely connected neighborhood has positive curvature and an edge serving as a local bridge has negative curvature. We use the edge curvature distribution to form a graph kernel which is then used to compare and cluster graphs. The curvature kernel uses purely the graph topology and thereby works for settings when node attributes are not available. The computation of the curvature for an edge uses only information within two hops from the edge and a random sample of $O(1/\varepsilon^2 \log 1/\varepsilon + 1/\varepsilon^2 \log 1/\delta)$ edges in a large graph can produce a good approximation to the curvature distribution with error bounded by ε with probability at least $1 - \delta$. Thus, one can compute the graph kernel for really large graphs that some other graph kernels cannot handle. This Ricci curvature kernel is extensively tested on graphs generated by different generative models as well as standard benchmark datasets from bioinformatics and Internet AS network topologies.

Graph classification and comparison are widely applied in bioinformatics, vision and social network analysis. One of the most popular approaches in practice is using graph kernels which compute the similarity of two graphs in terms of subgraph structures. Many graph kernels have been developed, which differ by the subgraph structures they focus on, such as random walks [2], shortest paths [1], subtrees [5], and cycles [3]. Graph kernels have been extensively tested on benchmark datasets from bioinformatics to chemistry [1, 4].

In our work, we focus on the setting of unlabeled graphs and propose a new graph kernel based on discrete Ricci curvature which takes only the network topology as an input. Our work is motivated by the use of curvature related kernels in shape matching. Curvature on a smooth surface defines the amount by which a geometric object deviates from being flat or straight. It is a local measure at each point but nevertheless has deep connections to global topology and structures. Despite the success in shape matching, curvature has not been used much for comparing graphs. In this paper, we propose to use curvature distribution of graphs to build new graph kernels. The goal is to demonstrate

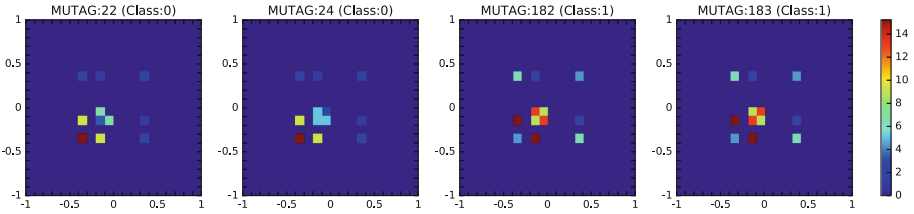


Fig. 1. The 2D Ricci Curvature histogram of MUTAG graphs which describes the curvature distribution for pairs of neighboring edges. A suitable choice of mass distribution on the neighborhood results in curvature always ranging between $[-1, 1]$. Graphs in the same class tend to have similar histogram distributions. Here MUTAG:22 and MUTAG:24 belong to the same class, while MUTAG:182 and MUTAG:183 belong to another class.

that the Ricci curvature distribution and kernels can be efficiently computed and capture interesting graph properties. They add to the family of graph features and kernels, could be combined with other attributes, and used for other classifiers.

2 Discrete Ricci Curvature

For an edge \overline{uv} , define a distribution m_u, m_v on the neighborhood of u, v respectively (such as uniform m_u and m_v). Now compute the Earth Mover Distance $W(u, v)$ from m_u to m_v , where the cost of moving mass from a neighbor u_i of u to a neighbor v_j of v is the shortest path distance in the graph. Here the edges are unweighted unless they inherit weights arising from the application domain, such as tie strength or distances. For example, $W(u, v)$ will be upper bounded by 2 for unweighted graph if we allocate 50% mass to the node’s neighbor. The Ollivier-Ricci curvature is defined as $w(\overline{uv}) = 1 - W(u, v)/d(u, v)$, where $d(u, v)$ is the length of edge \overline{uv} . Intuitively the curvature captures the structural properties of the local neighborhood. If \overline{uv} stays in a well connected, dense neighborhood, the curvature is positive; if \overline{uv} is locally a bridge, its curvature is negative.

3 Ricci Curvature Graph Kernel

We define the Ricci Curvature Kernel as the following. Denote the curvature distribution of all edges in G by $D(G)$ and that of G' by $D(G')$. We use the standard Gaussian RBF kernel: $k(G, G') = \exp(-\|D(G) - D(G')\|_2^2 / 2\sigma^2)$, where $\|D(G) - D(G')\|_2$ is the ℓ_2 norm of two vectors $D(G), D(G')$. Since the kernel depends on the curvature distribution, the distribution is less robust statistically for small graphs. We could boost up the kernel by considering the curvature distribution for pairs of neighboring edges $\{(w(e), w(e'))\}$. It appears to be more effective in practice. See Fig. 1 for an example. When the graph is really large, computing the curvature distribution might be costly ($O(|G.E| * n^3)$ where n is

the size of concerned neighborhood). Random sampling can be used to approximate the curvature distribution. Taking $O(1/\varepsilon^2 \log 1/\varepsilon + 1/\varepsilon^2 \log 1/\delta)$ edges uniformly at random from the graph G , it can be shown that $\hat{D}(G)$, the curvature distribution on the sampled edges, is a good approximation of $D(G)$ with error bound ε with probability $1 - \delta$. Notice that the running time does not even depend on the size of the graph n .

References

1. Borgwardt, K.M., Kriegel, H.P.: Shortest-path kernels on graphs. In: Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM 2005), pp. 74–81. IEEE Computer Society, Washington, DC (2005). <http://dx.doi.org/10.1109/ICDM.2005.132>
2. Gärtner, T., Flach, P., Wrobel, S.: On graph kernels: hardness results and efficient alternatives. In: Schölkopf, B., Warmuth, M.K. (eds.) COLT-Kernel 2003. LNCS (LNAI), vol. 2777, pp. 129–143. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-45167-9_11
3. Horváth, T., Gärtner, T., Wrobel, S.: Cyclic pattern kernels for predictive graph mining. In: Kim, W., Kohavi, R., Gehrke, J., DuMouchel, W. (eds.) Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004), 22–25 August 2004, Seattle, WA, USA, pp. 158–167. ACM Press, New York (2004). <http://doi.acm.org/10.1145/1014052.1014072>
4. Ralaivola, L., Swamidass, S.J., Saigo, H., Baldi, P.: Graph kernels for chemical informatics. *Neural Netw.* **18**(8), 1093–1110 (2005). <http://dx.doi.org/10.1016/j.neunet.2005.07.009>
5. Ramon, J., Gärtner, T.: Expressivity versus efficiency of graph kernels. In: Raedt, L.D., Washio, T. (eds.) Proceedings of the First International Workshop on Mining Graphs, Trees and Sequences (MGTS 2003) at the 14th European Conference on Machine Learning and 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD 2003), 22 and 23 September 2003, Cavtat-Dubrovnik, Croatia. pp. 65–74 (2003)

New Spectral Sparsification Approach for Drawing Large Graphs

Jingming Hu^(✉) and Seok-Hee Hong

The School of Information Technologies,
The University of Sydney, Camperdown, Australia
jihu2855@uni.sydney.edu.au, seokhee.hong@sydney.edu.au

1 Introduction

Nowadays many big complex networks are abundant in various application domains, such as the internet, finance, social networks, and systems biology. Examples include web graphs, AS graphs, Facebook networks, Twitter networks, protein-protein interaction networks and biochemical pathways. However, computing good visualization of big complex networks is extremely challenging due to scalability and complexity.

Recent work for visualizing big graphs uses a *proxy graph* approach [3]: the original graph is replaced by a proxy graph, which is much smaller than the original graph. The challenge for the proxy graph approach is to ensure that the proxy graph is a *good representation* of the original graph.

Eades *et al.* [2] presented proxy graphs using the spectral sparsification approach. *Spectral sparsification* is a technique to reduce the number of edges in a graph, while retaining its structural properties, introduced by Spielman *et al.* [5].

More specifically, they present a method for computing proxy graphs, called DSS (Deterministic Spectral Sampling), by selecting *edges* with high *resistance values* [5]. Their experimental results confirmed the promises by Spielman *et al.*: i.e., the spectral sparsification based methods are more effective than *Random Edge sampling* based method.

It was left as an open problem to compare the spectral sparsification based proxy graph approach with other graph sampling based proxy graph methods.

2 Our Results

In this poster, we introduce a new method called Spectral Sparsification Vertex (SSV-I) for computing proxy graphs using the spectral sparsification approach. Roughly speaking, we define resistance values for *vertices*, using the sum of resistance values of incident edges then we select vertices with high resistance values.

Suppose that $G = (V, E)$ is a graph with a vertex set V ($n = |V|$) and an edge set E ($m = |E|$). Let $r(v)$ represents a resistance value of a vertex v , and $r(e)$ represents a resistance value of an edge e . Let $deg(v)$ represents a degree of

a vertex v (i.e., the number edges incident to v), and let E_v represents a set of edges incident to a vertex v .

More specifically, we define *resistance value* for each vertex v as below:

$$r(v) = \sum_{e \in E_v} r(e)$$

We now describe a new method called SSV-I (Spectral Sparsification Vertex) for computing spectral sparsification based proxy graph $G' = (V', E')$ of $G = (V, E)$. Let V' consist of the n' of largest effective resistance. Then G' be the subgraph of G induced by V' .

Our experimental results with both benchmark real-world graphs and synthetic graphs using graph sampling quality metrics, visual comparison with various graph layouts and proxy graph quality metrics [3] show significant improvement by the SSV-I method over the Random Vertex (RV) sampling method.

Our main contribution and findings can be summarised as follows:

1. We introduced a new method called Spectral Sparsification Vertex (SSV-I) for computing proxy graphs using the spectral sparsification approach.
2. Experimental results with sampling metrics confirm that the SSV-I shows significant improvement over RV method. To be precise, around 35% improvement SSV-I over RV method on average in most metrics.
3. We observed that the Backbone layout [4] shows better structure for Benchmark graphs (i.e., real-world data), esp. scale-free graphs, and the Organic layout [1] produces better shape for Black-hole graphs (i.e., synthetic graphs).
4. Visual comparison of proxy graphs computed by SSV-I and RV using Benchmark, GION, and Black-hole data sets using the Backbone and Organic layouts confirms that our new SSV-I method produces proxy graphs with better connectivity structure with similar visual structure to the original graph than RV.
5. Experimental results confirm our hypothesis that the SSV-I method performs better than RV in proxy quality metrics, esp., when the relative density is low.
6. We observed that the Backbone layout performs better than the Organic layout in terms of the improvement in proxy quality metrics computed by SSV-I over RV method.

References

1. yEd - Java Graph Editor. <https://www.yworks.com/products/yed>
2. Eades, P., Nguyen, Q., Hong, S.-H.: Drawing big graphs using spectral sparsification. In: Frati, F., Ma, K.-L. (eds.) GD 2017. LNCS, vol. 10692, pp. 272–286. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73915-1_22
3. Nguyen, Q.H., Hong, S.H., Eades, P., Meidiana, A.: Proxy graph: visual quality metrics of big graph sampling. *IEEE Trans. Vis. Comput. Graph.* **23**(6), 1600–1611 (2017)

4. Noca, A., Ortmann, M., Brandes, U.: Untangling the hairballs of multi-centered, small-world online social media networks. *J. Graph Algorithms Appl.* **19**(2), 595–618 (2015)
5. Spielman, D.A., Teng, S.H.: Spectral sparsification of graphs. *SIAM J. Comput.* **40**(4), 981–1025 (2011)

SPQR Proxy Graphs for Visualization of Large Graphs

Seok-Hee Hong^(✉) and Quan Nguyen

The School of Information Technologies,
University of Sydney, Camperdown, Australia
quan.nguyen@sydney.edu.au

1 Introduction

Recent work for visualizing large graphs uses a *proxy graph* method [3]: the original graph is replaced by a proxy graph, which is much smaller than the original graph. The challenge for the proxy graph approach is to ensure that the proxy graph is a *good representation* of the original graph. However, previous work to compute proxy graphs using the *random sampling* methods often fail to preserve the important global skeletal structure and connectivity of the original graph [4, 5].

For example, Zhang *et al.* presented experimental comparison of different sampling algorithms under various sampling metrics [5]. Wu *et al.* presented user studies to investigate how sampling methods influence graph visualization, in terms of human perception of high degree vertices, clusters and coverage area; it was recommended to use Random Walk (RW) for high degree vertex, Random Jump for clustering, but to avoid Random Vertex (RV) sampling [4]. In particular, Random Vertex and Random Edge sampling often produce a set of disconnected proxy graphs [4].

The *BC (Block Cut-vertex) proxy graph* methods, based on the BC tree decomposition of a connected graph into biconnected components, produced better results than the random sampling based methods. However, the bottleneck was when graphs have giant biconnected components [2]. In particular, the performance gain for real-world graphs was smaller, due to the existence of single dominant component in real-world graphs.

Therefore, it was left as an open problem and future work is to conduct further experiments, by combining with other graph partitioning methods [2].

2 The SPQR Tree

The *SPQR tree* of a biconnected undirected graph G represents the decomposition of G into triconnected components [1], which can be computed in linear time.

We use basic terminology of SPQR trees; for details, see [1]. Each triconnected component consists of *real edges* (i.e., edges in the original graph) and *virtual*

edges. (i.e., edges introduced during the decomposition process, which represents the other triconnected components, sharing the same virtual edges defined by cut-pairs).

Each node ν in the SPQR tree is associated with a graph called the *skeleton* of ν , denoted by $\sigma(\nu)$, which corresponds to a triconnected component. There are four types of nodes ν in the SPQR tree: (i) S-node, where $\sigma(\nu)$ is a simple cycle with at least three vertices; (ii) P-node, where $\sigma(\nu)$ consists of two vertices connected by at least three edges; (iii) Q-node, where $\sigma(\nu)$ consists of two vertices connected by two (real and virtual) edges; and (iv) R-node, where $\sigma(\nu)$ is a simple triconnected graph with at least four vertices.

In this poster, we use the SPR tree, a simplified version of the SPQR tree *without* Q-nodes, since the Q-node consists of two vertices and edges.

3 Our Results

This poster introduces new *SPQR proxy graph* methods, integrating graph sampling methods with the *SPQR tree* [1] to maintain the important global connectivity structure of the original graph.

We present two new families of proxy graph methods **SPQR-W** and **SPQR-E**, each contains the five most popular sampling methods, including RV (Random Vertex), RE (Random Edge), IRE (Induced Random Edge), RP (Random Path) and RW (Random Walk), used in previous work [2–5].

More specifically, we first include the *separation pairs* of the original graph to proxy graphs, since separation pairs are structurally important vertices in terms of connectivity. Then, **SPQR-W** proxy graph methods perform sampling using the original sampling algorithms.

SPQR-E algorithm is a Divide and Conquer algorithm that uses **SPQR-W** algorithms: it first selects separation pairs, and then performs **SPQR-W** algorithms for each triconnected component $\nu_i, i = 1, \dots, k$ of G to compute a proxy graph G'_i of $\sigma(\nu_i)$, the *skeleton* of ν . Finally, it merges $G'_i, i = 1, \dots, k$, into the final proxy graph G' of G .

Note that the skeleton $\sigma(\nu)$ consists of virtual edges and real edges. Since such virtual edges do not exist in the original graph, we only sample real edges of $\sigma(\nu)$.

The main contribution of this poster is summarized as follows:

1. We present two new families of proxy graph methods **SPQR-W** (SPQR-Whole) and **SPQR-E** (SPQR-Each). Each family consists of five new methods, integrating the SPQR tree decomposition with the most popular five sampling methods, used in the sampling-based proxy graph method [2, 3].
2. Experimental results using graph sampling quality metrics, proxy quality metrics [3] and visual comparison with real world graphs show that our new SPQR proxy graph methods produce significantly better results than the previous methods [2, 3].

References

1. Di Battista, G., Tamassia, R.: On-line planarity testing. *SIAM J. Comput.* **25**(5), 956–997 (1996). <https://doi.org/10.1137/S0097539794280736>
2. Hong, S., Nguyen, Q., Meidiana, A., Li, J., Eades, P.: Bc tree based proxy graphs for visualization of big graphs. In: *IEEE PacificVis 2018*, pp. 11–20 (2018)
3. Nguyen, Q.H., Hong, S., Eades, P., Meidiana, A.: Proxy graph: visual quality metrics of big graph sampling. *IEEE Trans. Vis. Comput. Graph.* **23**(6), 1600–1611 (2017). <https://doi.org/10.1109/TVCG.2017.2674999>
4. Wu, Y., Cao, N., Archambault, D., Shen, Q., Qu, H., Cui, W.: Evaluation of graph sampling: a visualization perspective. *IEEE Trans. Vis. Comput. Graph.* **23**(1), 401–410 (2017)
5. Zhang, F., Zhang, S., Wong, P.C., Swan II, J.E., Jankun-Kelly, T.: A visual and statistical benchmark for graph sampling methods. In: *Exploring Graphs at Scale (EGAS) Workshop, IEEE VIS 2015*, October 2015

Taming the Knight's Tour: Minimizing Turns and Crossings

Juan Jose Besa, Timothy Johnson, Nil Mamano^(✉), and Martha C. Osegueda

Department of Computer Science, University of California, Irvine, CA, USA
{jjbesavi,tujohnso,nmamano,mosegued}@uci.edu

1 Introduction

The classic Knight's Tour Problem asks for a sequence of knight moves in an $n \times n$ chess board that allows the knight to visit every square exactly once and return to the starting position. There is a long history of algorithms for producing knight tours (e.g., see [1, 3]). However, most of them produce complex tours. We consider the problem of finding knight's tours minimizing two metrics of complexity: the number of turns and the number of crossings. A *turn* is when two consecutive knight moves in the tour go in different directions (i.e., when the three cells involved are not collinear); a *crossing* is when the line segment connecting the cells of two knight moves intersect. To the best of our knowledge, these metrics are new in this context, but they are often studied in geometric contexts and, in the case of crossings, in graph drawing. (However, people have looked at the related problem of the longest knight path without any crossings [2].)

We use a novel approach to produce a family of knight's tours for $n \times n$ boards (where n is even, since, otherwise, a tour does not exist) with a near-optimal number of turns and crossings (see Results). Our approach also has several other good qualities: **(i)** the knight move at any given cell can be determined in constant time without constructing the tour explicitly; **(ii)** it can be generalized to rectangular boards (as long as both sides do not have odd length, in which case a tour does not exist), and **(iii)** it the tours are easy to visualize and construct without the need of computers or calculations.

Results. Our tours have $10.75n + O(1)$ turns and $13n + O(1)$ crossings, where the constant factors are quite small but vary slightly depending on $n \bmod 8$. For instance, if $n \equiv 2 \pmod{8}$, the constants are 40.5 and 91, respectively. Since a knight must turn at any cell next to the edge of the board, *any* knight tour must have at least $4n$ turns.¹ Similarly, by examining the ways to cover the first two rows (or columns) on each side of the board, we can show that there is at least one crossing per column per side. Therefore, the number of crossings must be also be at least $4n$. Therefore (for sufficiently large values of n) our tours are within a factor of ≈ 2.7 and 3.25 of the minimum.

¹ This lower bound can be improved to $4.25n$ by observing that cells close to the center of the board must be part of a sequence of moves that must contain a turn that is not in one of the cells along the boundary. We omit this slight improvement for brevity.

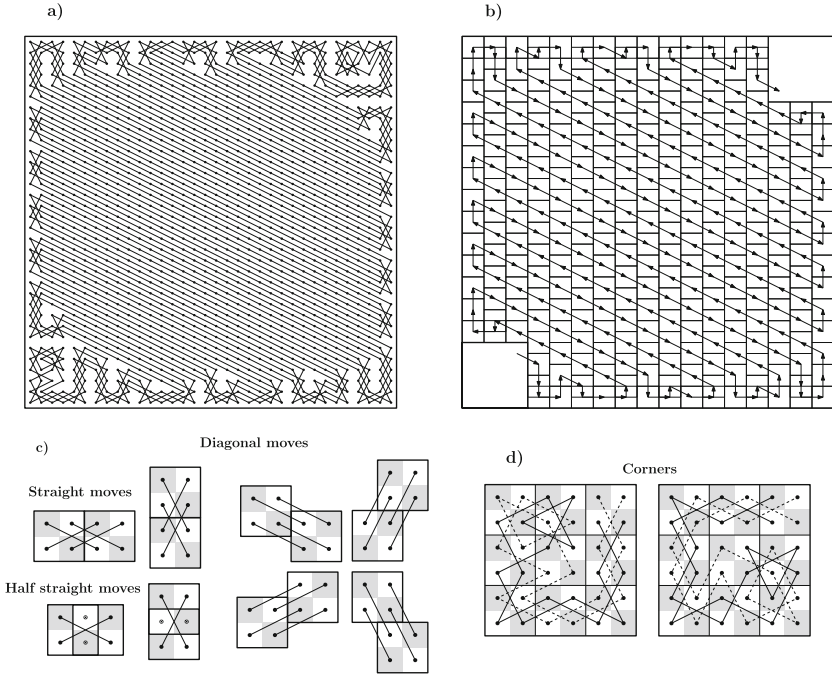


Fig. 1. (a) Knight’s tour on a 34×34 chess board with 320 turns and 429 crossings (Other dimensions at [4]). (b) Corresponding sequence of “formation moves”. (c) “formation moves” for a group of 2×2 knights. In half straight moves only two of the four knights move maintaining the 2×2 formation (d) corner constructions that pair up four knight paths into two connected paths (solid and dashed).

2 Construction

We begin by covering the board (except two corners) with four knights arranged in a 2×2 formation. By using the “formation moves” depicted in Fig. 1c, they can cover the board while remaining in formation (see Fig. 1b). The main idea is to move in zig-zag along diagonals, because diagonal moves do not create turns or crossings. A little care must be put when selecting the moves along the bottom and top edges of the board to make sure that every cell is visited; nonetheless, it is not hard to do so. The issue of how to transform the four knights into a single knight’s tour is resolved by using a special construction at the bottom-left and top-right corners of the board (see Fig. 1d). This construction pairs up the four knights into two connected paths at each of these two corners. This results in either a valid knight’s tour or two disjoint cycles. However, note that there are two alternative corners which pair up different knights; therefore, it is always possible to set the corners to make a valid knight tour.

References

1. Ball, W.W.R.: *Mathematical Recreations and Essays*. Macmillan (1914)
2. Knuth, D.E.: *Selected Papers on Fun and Games*. In: *CSLI Lecture Notes Series* (2011)
3. Parberry, I.: An efficient algorithm for the Knight's tour problem. *Discrete Appl. Math.* **73**(3), 251–260 (1997)
4. <https://www.ics.uci.edu/~nmamano/knightstour.html>

Author Index

- Akitaya, Hugo A. 361
Alam, Jawaherul Md. 213
Angelini, Patrizio 67, 123, 495, 630
Argyriou, Evmorfia 509
Arroyo, Alan 649
Arseneva, Elena 371
- Bachmaier, Christian 646
Barth, Lukas 624
Bekos, Michael A. 77, 123, 213, 271, 495, 630
Besa, Juan Jose 661
Bleuse, Raphaël 106
Bose, Prosenjit 371
Bouvry, Pascal 106
Brückner, Guido 39, 624
- Camacho, Charles 633
Cano, Pilar 371
Castermans, Thom 53
Chaplick, Steven 137, 636
Chen, Hang 463
Chimani, Markus 255
Cornelsen, Sabine 509
Cortese, Pier Francesco 23
- D'Angelo, Anthony 371
Da Lozzo, Giordano 92
Danoy, Grégoire 106
De Luca, Felice 303, 433
Demel, Almut 286
Deniz, Zakir 317
Devanny, William 609
Di Battista, Giuseppe 92
Di Giacomo, Emilio 77
Didimo, Walter 77, 481, 495, 621
Döring, Hanna 255
Dujmović, Vida 371
Dürschnabel, Dominik 286
- Eades, Peter 67, 569
Eppstein, David 541
- Fabila-Monroy, Ruy 593
Felsner, Stefan 555
- Fernández-Merchant, Silvia 633
Firman, Oksana 643
Förster, Henry 123, 271, 509
Frati, Fabrizio 92, 371
Fulek, Radoslav 229
- Galby, Esther 317
Galvão, Marcelo 627
Gao, Jie 447, 652
Goodman, Dan F. M. 640
Geckeler, Christian 271
Grilli, Luca 495, 621
Gronemann, Martin 213
Gu, Xianfeng 447
- Healy, Patrick 569
Hidalgo-Toscano, Carlos 593
Holländer, Lukas 271
Hong, Seok-Hee 67, 655, 658
Hossain, Md Iqbal 303
Hu, Jingming 655
Huemer, Clemens 593
- Jelic, Marija 633
Johnson, Timothy 661
Jungeblut, Paul 624
Jünger, Michael 187
- Kaufmann, Michael 123, 213, 242, 271, 509, 630
Kieffer, Emmanuel 106
Kindermann, Philipp 152, 495, 609
Kirsch, Rachel 633
Klammler, Moritz 169
Klein, Karsten 67
Kleist, Linda 333, 633
Knauer, Kolja 200
Kobourov, Stephen 67, 303, 433, 463
Krukar, Jakub 627
- Langerman, Stefan 371
Lara, Dolores 593
Lin, Yu-Yao 447, 652
Liotta, Giuseppe 67, 77, 481, 524, 621

- Lipp, Fabian 137, 643
 Liu, Kin Sum 652
 Löffler, Andre 636
 Löffler, Maarten 361, 609
 Lu, Yafeng 463
 Lubiw, Anna 303, 387

 Maciejewski, Ross 463
 Mamano, Nil 661
 Matson, Elizabeth Bailey 633
 Mchedlize, Tamara 169, 286, 416, 495
 Meder, Jeff 106
 Meulemans, Wouter 53
 Micek, Piotr 200
 Miltzow, Tillmann 387
 Mitsche, Dieter 593
 Mondal, Debajyoti 303, 387
 Montecchiani, Fabrizio 77, 152, 524, 621
 Munaro, Andrea 317
 Mütze, Torsten 354
 Mutzel, Petra 187

 Navarra, Alfredo 67
 Nguyen, Quan 658
 Ni, Chien-Chun 447, 652
 Nikolov, Nikola S. 569
 Nöllenburg, Martin 53, 509

 Oikonomou, Anargyros 347
 Okamoto, Yoshio 509
 Olszewski, Maya 106
 Ortali, Giacomo 579
 Osegueda, Martha C. 661

 Pach, János 242
 Pagliuca, Daniele 621
 Pak, Alexey 169
 Parada, Irene 361
 Patrignani, Maurizio 23, 92, 481
 Pawar, Samraat 640
 Prutkin, Roman 495
 Pupyrev, Sergey 213
 Purchase, Helen 433

 Radermacher, Marcel 286, 402, 624
 Raftopoulou, Chrysanthi 509
 Reitzner, Matthias 255
 Richter, R. Bruce 649
 Ries, Bernard 317
 Rosalie, Martin 106
 Roselli, Vincenzo 92
 Rutter, Ignaz 39, 402, 609, 646

 Scheucher, Manfred 354, 555
 Schlipf, Lena 152
 Schmöger, Rainer 636
 Schneck, Thomas 630
 Schulz, André 152
 Schwering, Angela 627
 Soni, Utkarsh 463
 Spallek, Amadäus M. 271
 Spisla, Christiane 187
 Splett, Jan 271
 Straube, Laura 643
 Stumpf, Peter 39, 646
 Sunohara, Matthew 649
 Symvonis, Antonios 347, 495

 Tappini, Alessandra 67, 371, 495, 524
 Telea, Alexandru 3
 Tollis, Ioannis G. 579
 Tóth, Csaba D. 229
 Tóth, Géza 242

 Ueckerdt, Torsten 200, 242
 Urhausen, Jérôme 416

 van Garderen, Mereke 53

 White, Jennifer 633
 Wolff, Alexander 137, 509, 643
 Wulf, Lasse 286

 Yuan, Xiaoru 53

 Zheng, Jonathan X. 640
 Zink, Johannes 137