Graph Peeling Semantics

James Abello[1,2], Haoyang Zhang[1]

[1]Computer Science Department, Rutgers, The State University of New Jersey
[2]DIMACS, Rutgers, The State University of New Jersey

Abstract

Recently, Graph Cities [1, 2] have been proposed as scalable 3d visual representations of graph edge partitions where each subgraph in the partition is a "fixed point of degree peeling". In this work, we propose "intuitive" primitives to extract language semantics from the topology of these fixed points aided by provided graph vertex labels. The main approach is to view the collection of data labels as a set system derived from the graph topology and to derive "intuitive" language semantics from a specially derived set system intersection meta-graph. Exploration primitives include a glyph grid map of the distribution of all fixed points in the data set and a textual summary tool. We illustrate our approach with a variety of fixed points subgraphs extracted from "large" datasets that include a patent citation network (16.5 million edges) [3], a movie keywords co-occurrence network derived from the Internet Movie Database (5 million edges), a paper citation network derived from arXiv Computer Science papers (1.5 million edges), and a Parler dataset [4].

Keywords

Information Interfaces and Presentation (e.g., HCI), Data Structures, Analysis of Algorithms and Problem Complexity, Graph Theory, Information Search and Retrieval, Computer Graphics, Massive Datasets

1. Introduction

The proliferation of "large" data sets has made the search for scalable interactive visual representations an area of pressing importance. Two of the main questions in this type of investigation relate to the Screen and I/O Bottlenecks [5, 6]. The I/O bottleneck refers to the fact that specialized "external memory" algorithms [7] are required to process graphs when their edge set resides on disk but the full adjacency list does not fit in the computer's RAM. The screen bottleneck emphasizes the need to devise visual representations that are aware of the computer screen size at different levels of resolution and with different user interactivity requirements. The overall goal is to amplify the user's understanding of the essential properties of a non-RAM resident graph by offering exploration and summarization tools whose combination becomes a "graph sense making" machinery.

We invite the reader to consider graphs with several billion edges (i.e., GigaGraphs) residing on files on a computer disk with 32 GB of RAM for processing. Can we devise a "small" visual representation that can be explored interactively at different levels of resolution and that can be used to generate a "summary" of selected graph properties? Ideally, the visual representation should be suitable to be used as a "visual stamp" of the overall network "structure". *Graph Cities* constitute an example of such "small" visual representations.

These representations at the end of the day need to provide users "intuitive" mechanisms to select data regions at different levels of granularity in order to produce summaries that incorporate information derived from the labels associated with the graph data. We report on our current attempts to provide some algorithmic primitives that use the layered topology of fixed points to aggregate vertex labels into a hierarchical set system from which summaries of the fixed point contents can be obtained. Our emphasis here is on the algorithmic primitives applied to small subgraphs of fixed points extracted from very large graph data. Our intention is to provide "intuitive" data summaries derived from the vertex labels. Extraction of global-level semantics from billion-edge graphs is a work in progress. Here we focus on specially selected fixed points.

1.1. Related Work

The choice of representations that facilitate smooth visual interaction is a subject of active research [8, 9, 10]. All these previous techniques have algorithms with running time *substantially greater than linear* on the number of graph elements, making them not suitable for massive graph visualization. Graph Thumbnails, as a mechanism to identify and compare multiple graphs, are alone the subject of [11]. Generation of graphs with a predefined core structure is the focus of [12]. Computational aspects of core related graph decompositions and graph sparsification are studied in [13, 14, 15, 16, 17]. Machine learning approaches, such as those described in [18, 19], have been proposed to learn low-level embeddings of graphs, however, such approaches are not yet suitable

Proceedings of the 6th International Workshop on Big Data Visual Exploration and Analytics co-located with EDBT/ICDT 2023 Joint Conference (March 28-31, 2023), Ioannina, GR
abelloj@cs.rutgers.edu (J. Abello); hz333@rutgers.edu (H. Zhang)

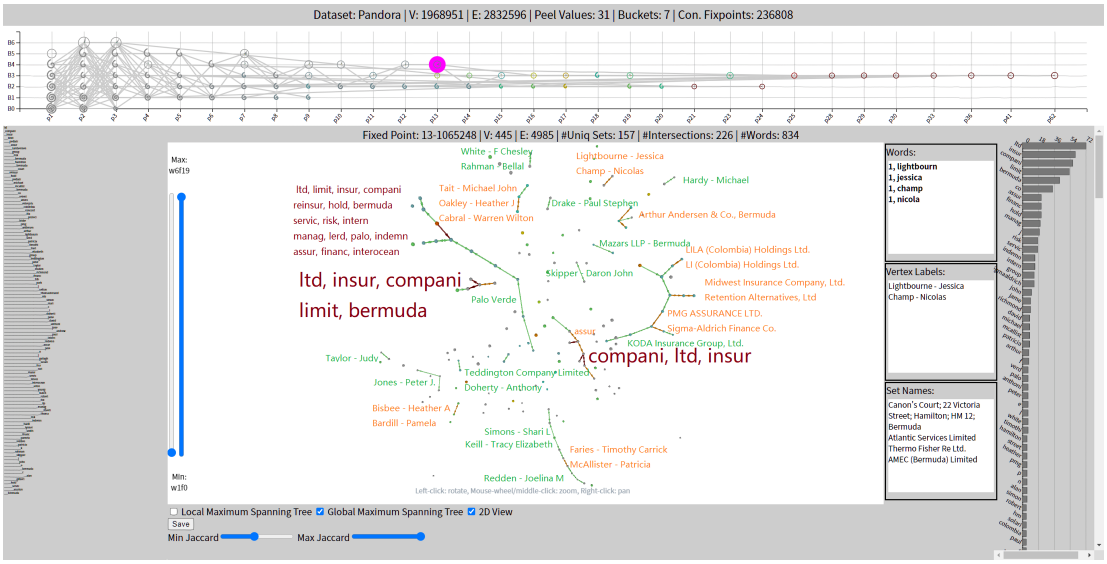

Figure 1: Set system intersection meta-graph computed for a fixed point F_{13} (in red) from a Pandora Papers dataset. A quick search engine query verifies that some of the detected company names have been dissolved, for example, Teddington Company Limited.

for the extraction of label semantics from billion-edge graphs.

1.2. Contribution

We built on previous work on Graph Cities [1, 2] to select fixed points of degree peeling for extraction of semantics from their vertex labels. We rely on the waves and fragments decomposition of [20] to hierarchically aggregate the fixed point vertex labels into a set system whose pairwise intersections are used to compute a maximum spanning tree from which a hierarchical summary of the fixed point labels is offered to the user for further exploration. Concretely, our contributions are:

1. A Visual interface (Figure 1) for hierarchical labels summary of the fixed point vertex labels.
2. A glyph map (Figure 2) that is used as a visual index for fixed point selection.
3. Editing facilities for users to select and annotate subgraph patterns of his/her own interests that can be incorporated into graph city galleries

The paper layout is as follows: Section 2 presents the details of global and local views of fixed points as directed meta-graphs and introduces a glyph map that provides direct access to any fixed point. Section 3 lays out the construction of a set system of labels derived from the layered topology of the input fixed point, and provides sample results from a citation network [21, 22], the Pandora papers[1], a movie keyword co-occurrence dataset[2], and the Parler dataset [4]. Section 4 discusses our current work directions on related open problems. Section 5 concludes the paper.

Even though our current results are preliminary they are very encouraging and we will be focusing in the near future on the creation at a scale of the fixed point semantic approach reported here.

2. Global and Local Views of Fixed Points

2.1. Fixed Points Intersection Set System

The fundamental graph theoretical building units used in this work are fixed points F_k of minimum degree peeling k [23]. The reason is that the edges of any graph can be partitioned into maximal subgraphs that are Fixed Points. To our knowledge, the best algorithm for the iterative edge decomposition of any graph into fixed points has complexity $O(\sqrt{|E|}|E|)$ and this follows from the fact that a graph cannot have more than $\sqrt{|E|}$ different peel values. Figure 4 depicts a small fixed point. It consists of a sequence of waves, and each wave consists of an ordered sequence of edge sets called edge fragments adjacent to disjoint sets of vertices called layers.

[1] https://www.icij.org/investigations/pandora-papers/
[2] https://www.imdb.com/interfaces/

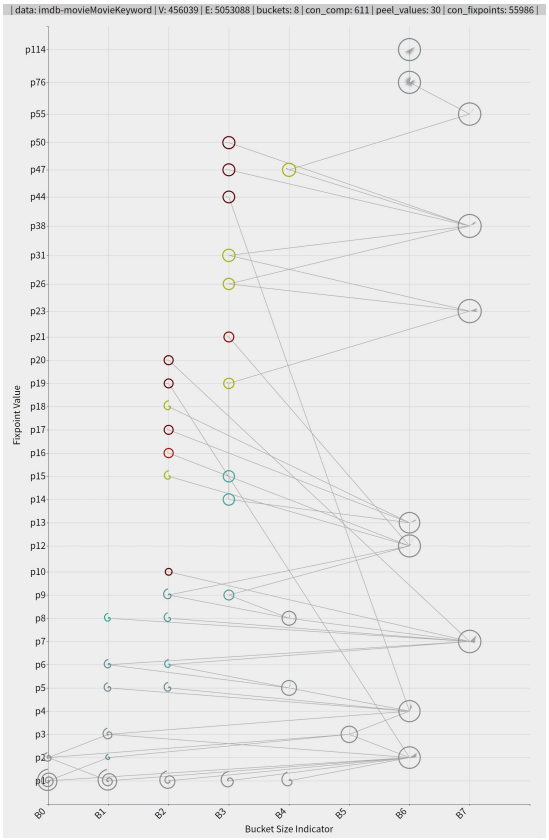

Figure 2: Glyph map of movies to movies and keywords network, where the x-axis encodes fixed point size buckets, and the y-axis represents fixed point values. Each glyph represents a collection of fixed points with the same fixed point value and size within a logarithmic factor.

An alternative view of an overall edge partition of a graph into fixed points is to consider the intersection meta-graph determined by the vertex sets of the fixed points, i.e. two fixed points $< F_i, F_k >$ are connected if their vertex sets intersect and their connecting edge is weighted by the cardinality of their intersection divided by the size of their union. A spanning directed subgraph view of this fixed point intersection meta-graph can be obtained by collapsing each connected fixed point into a meta-node and directing all edges from lower fixed point values to higher fixed point values (Figure 2). This is a **D**irected **A**cyclic **M**eta-**G**raph (DAMG) view of the fixed point decomposition of the entire dataset.

Fixed Points Glyph Grid Map. To provide easy access to any fixed point subgraph of a graph city, a grid map is provided, where each point is addressable by a bucket size indicator (x-axis) and a fixed point value (y-axis) (see Figure 2).

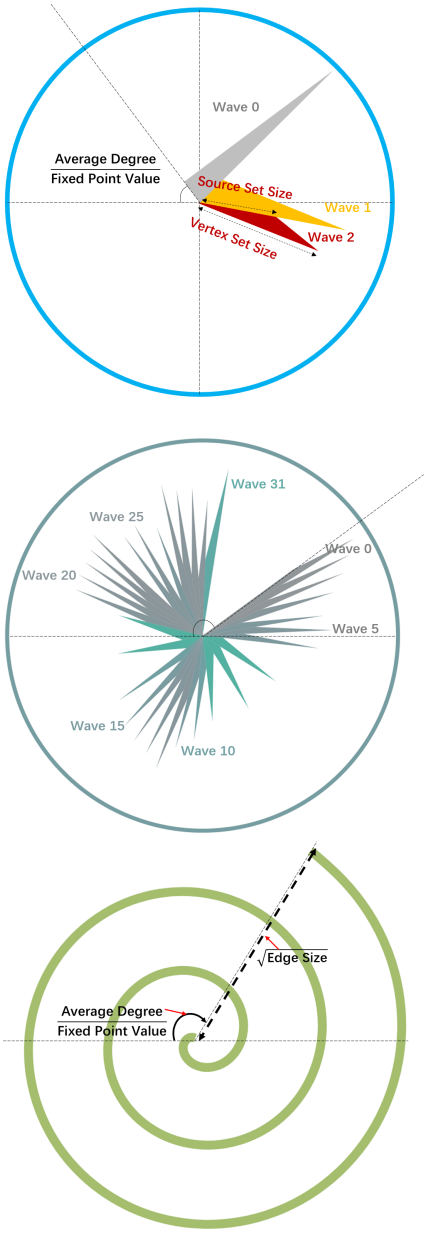

Figure 3: The top figure depicts a circular glyph representing a connected fixed point. The internal triangle spikes encode its waves. The Middle figure is an example of a circular glyph representing fixed point F_{16} in the Parler dataset. It has the largest number of waves. The bottom figure is a spiral glyph example summarizing a collection of fixed points with the same fixed point value and sizes within a logarithmic factor.

Each point in this grid has an associated glyph (see Figure 3) that summarizes the collection of the corresponding connected fixed points with the same fixed point value and similar edge size within a logarithmic factor (i.e., the same bucket). A circular glyph represents a fixed point, whose area and color of the ring encode the fixed point edge size and average density, respectively. Internally, a circular glyph contains a clockwise sequence of spikes that corresponds to the sequence of waves of the corresponding fixed point. The number of spikes is equal to the number of waves. Each spike corresponds to a triangle that encodes the wave seed set and the number of wave edges. The wave density is color encoded. The starting angle from the left to the first spike encodes the ratio between the fixed point's average degree and its peel value. When a set of fixed points have the same peel value and similar sizes within a logarithmic factor, a spiral glyph summarization represents the edge size and density of the union by its area and color. The spiral length encodes the number of fixed points in the collection, and the start angle from the left to the outer end represents the ratio between the average degree of the union and the peel value.

Since a glyph grid map provides a 2-dimension summary for a graph city, it can be used as a selector for "interesting" fixed points. (See Figure 3 (middle).) Hovering on a particular glyph in the grid map displays some statistics of the corresponding fixed points including the number of vertices, and edges.

Users can filter "interesting" fixed points by querying for the largest, densest, or most diverse.

2.2. Fixed Points as Sequence of Waves and Fragments

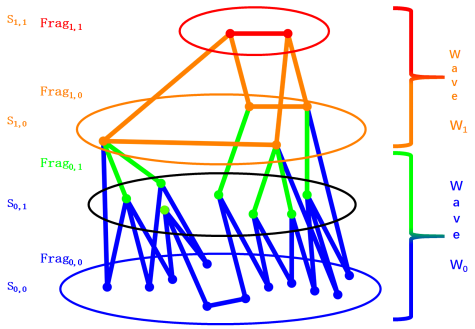

Figure 4: A fixed point can be viewed as an ordered sequence of waves, and each wave is a sequence of fragments

Edge fragments. Given a seed subset S of vertices of a graph, its edge fragment Fragment(S) consists of all those edges with at least one endpoint in S (i.e. edges touching S). Those endpoints of edges in Fragment(S) that are not in S are called the Boundary vertexes of Fragment(S).

A fixed point as a stack of edge fragments. A fixed point F_k can be viewed as a directed graph by iterative restricted exploration of its edge fragments as follows (see Figure 4). Take all vertexes of degree k as its seed source set, mark them as visited, and mark all edges "touching" this source set as visited. Update the degrees by subtracting from each vertex the number of newly visited edges adjacent to it, i.e. update the leftover out degree. Proceed iteratively by visiting in parallel the neighborhoods of those boundary vertexes whose leftover out-degree is strictly less than k. If there are no boundary vertices adjacent to the set of visited vertices with leftover degrees less than k it means that a new seed set of leftover degrees exactly equal to k must be used for further exploration. This is an indication of the beginning of a new subgraph of the fixed point that is generated by a new seed set of leftover out-degree k. (See next subsection). These maximal subgraphs generated by maximal disjoint seed sets of vertices of minimum degree k are what are called Graph Waves in [20]. In summary, Fixed points of peel value k contain maximal subsets of "seed" vertices of degree k that generate edge disjoint maximal subgraphs called the waves associated with the seed sets. Each of these waves has a beginning and an ending set of vertices with some collection of paths interconnecting them [1, 2].

Meta-DAG View of Fixed Points. Associated with the sequence of edge fragments forming a wave, their seed sets form an ordered partition of the vertex set of the wave. We direct edges according to the fixed point seed set ordering. An edge with both endpoints in the same seed set is called local, and edges directed from the lower seed set to the higher seed set are called out-going edges, otherwise, they are called in-coming edges. The number of outgoing edges incident to a vertex is called its left-over degree. With these preliminaries, we can introduce a **D**irected **A**cyclic **M**eta-**G**raph (DAMG) view of a fixed point as follows: The connected components of the subgraphs induced by the seed sets become meta-notes. Edges running from a connected component C in a seed set to a connected component D in a different seed set are aggregated as a directed meta-edge $(C, D, |E(C, D)|)$, where $|E(C, D)|$ encodes the number of directed edges running from C to D.

3. Generating Fixed Point Summaries

3.1. Generation of Hierarchical Summaries

Figure 5 illustrates the interface layout. We refer to the middle area of the screen as the canvas. On the top of

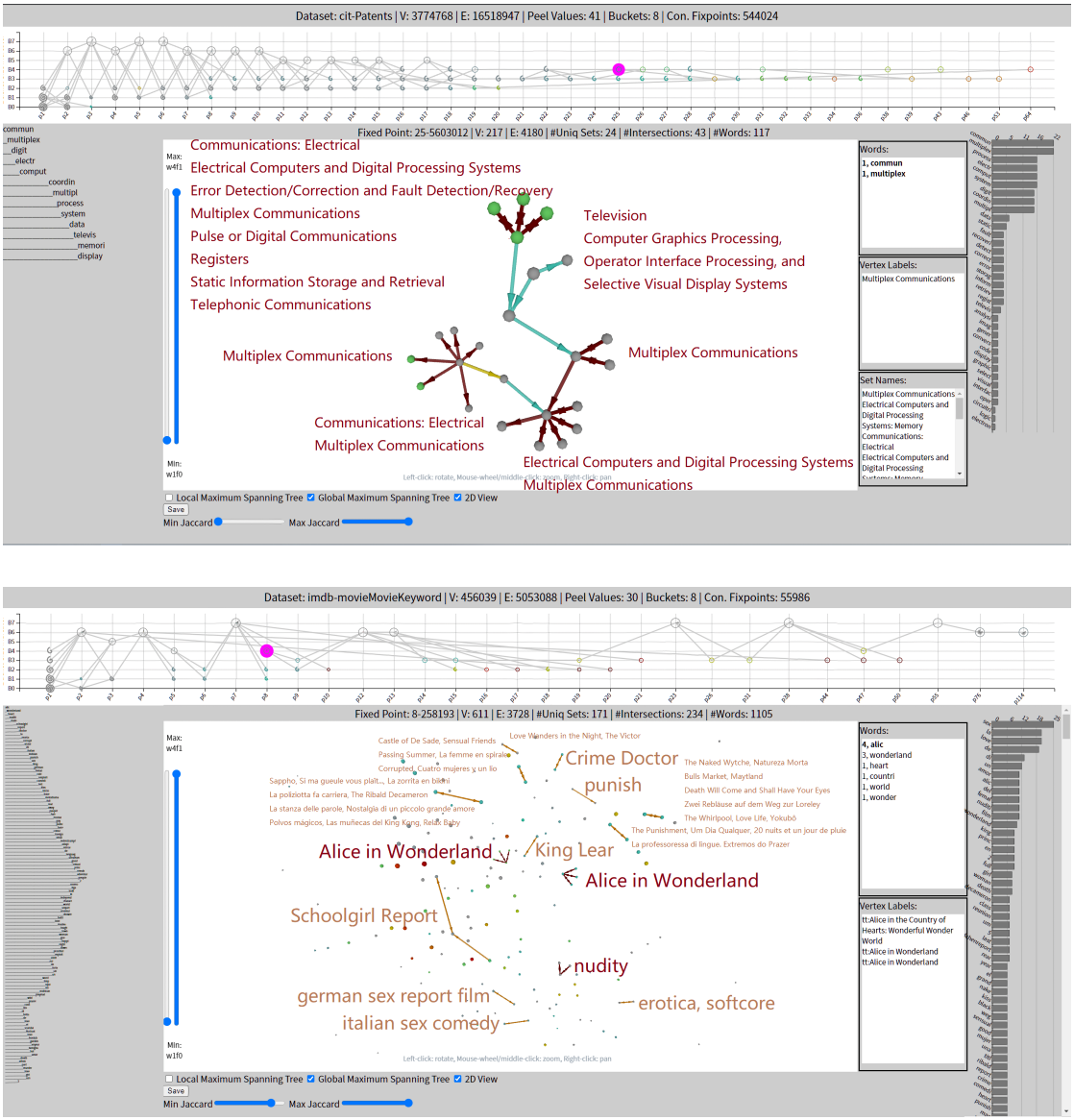

Figure 5: Top figure: Set system intersection meta-graph computed for a fixed point F_{25} from the US patent citation network. The displayed labels have a higher (above the mean) co-occurrence frequency in the labels set system. The selected categories of patents have a high level of citations to the other selected categories. Bottom figure: Set system intersection meta-graph computed for a fixed point F_8 from the movies to movies and keywords network. The displayed labels have a higher (above the mean plus one standard deviation) co-occurrence frequency in the labels set system. Selected movie titles are "similar" with respect to their viewers' keyword descriptions. Unfortunately, some of the keyword descriptions are very loose, and this produces a possible misclassification. For example, "Alice in Wonderland" appears here because the data contains the keyword "based on novel".

the canvas, there is a grid map from which users can select a fixed point (in red). To the left, a textual tree of hi-frequency labels is derived from a binary tree traversal as explained below. In the canvas of the screen, we display a maximum spanning tree of the label set system intersection meta-graph. When a user hovers over a link, its corresponding labels are displayed in an infobox. On the right-hand side, a label frequency bar chart is

displayed and interactively updated according to users' desired specifications.

The complexity of generating summaries is $O(|V|^2)$, where $|V|$ is the number of vertices in the fixed point. Please notice that non-quadratic implementations are possible by using Locally Sensitive Hashing [24].

Assuming that the vertices of a fixed point are labeled by sets of words, we describe next how we rely on the layered topological view of a fixed point to generate a set system of labels whose intersection graph is used to generate a hierarchical summary of the overall fixed point vertex labels.

Generating a set system of labels from the vertex labels. Denote by $F(k,h)$ a vertex labeled fixed point F with peel value k consisting of h fragments with a corresponding ordered sequence of seed sets $S_0, S_1, ..., S_{h-1}$ according to their fragment indices $frag_0, frag_1, ..., frag_{h-1}$. We output a hierarchical summary of the overall fixed point set of labels by building a bottom-up aggregation of the vertex labels in the non-decreasing order of fragment indices. Initially, we compute the connected components of the subgraph induced by the seed set S_0 and assign to each such component the union of the sets of labels of its vertices. We derive a Label Set System from the fixed point layered view decomposition as follows: Bottom-up, for each vertex $x \in frag_i, i = 0, 1, ..., h-1$, update the label set of x as the union of all its incoming neighbors' label sets. For each pair of vertices (u, v) from different fragments, if there exists a directed upward path from u to v, then connect (u, v) with a semantic edge weighted as the cardinality of the labels' set intersection between u and v. Call this graph *IntersectionLabelSetSystem($F(k,h)$)*. From this intersection label set system, we extract a summarization based on maximum spanning trees and a binary tree traversal. Visually, we build a color map according to the distribution of weights in the maximum spanning tree of the *IntersectionLabelSetSystem($F(k,h)$)*. A textual summary is extracted by a binary tree traversal of the maximum spanning tree according to the non-increasing order of weights, and we select from each tree edge being visited a label that has not been seen during the traversal.

3.2. Sample Results of Hierarchical Labeling

We illustrate the hierarchical labeling results obtained for fixed points selected from a patent citation network (Figure 5 (top)), a movies-to-movies-and-keywords dataset (Figure 5 (bottom)), a paper citation network (Figure 6 (top)), and the Parler dataset (Figure 6 (bottom)). The displayed labels are those that have a "substantial" number of co-occurrence in the set system of label sets.

4. Current and Future Work

We are currently designing navigation and summarization tools so that a user can annotate those subgraph patterns that he/she finds interesting. Currently, a user can annotate subgraph patterns indicating his findings and adding the corresponding subgraph patterns to a pattern gallery accessible from the top of the user interface. A fisheye view on the textural tree

4.1. Open Problems

1. What is the I/O complexity of computing the edge decomposition of a fully external memory graph? Namely, neither the vertices nor the edges fit in RAM.
2. Is there an efficient method to compute the fixed point edge decomposition in a streaming fashion?
3. What are examples of graph computations whose solutions can be obtained as compositions of their solutions on the graph fix points?
4. The decomposition of the edges of a graph into fixed points defines intrinsically an intersection graph of the collection of sets of vertices appearing on each fixed point. What are the properties of the graphs that are the intersection graphs of fixed point edge graph decompositions?

5. Conclusions

Devising visual representations of very large graphs is a tantalizing area of research. Coming up with tools that "explain" the semantics encoded by graph data labels at different levels of granularity is in our view a complementary endeavor that deserves more attention from the community. It opens avenues of interdisciplinary work involving at least natural language processing (textual and semantic similarity [25]), computer human interaction, and machine learning. We hope this modest contribution entices other researchers to join us in this quest.

6. Acknowledgements

This work was partially supported by NSF grants IIS-1563816, IIS-1563971, and mgvis.com. Thanks to the DIMACS staff for their support, and to Prof. Tim Tangherlini and Dr. Peter Broadwell for providing a curated version of the Parler dataset.

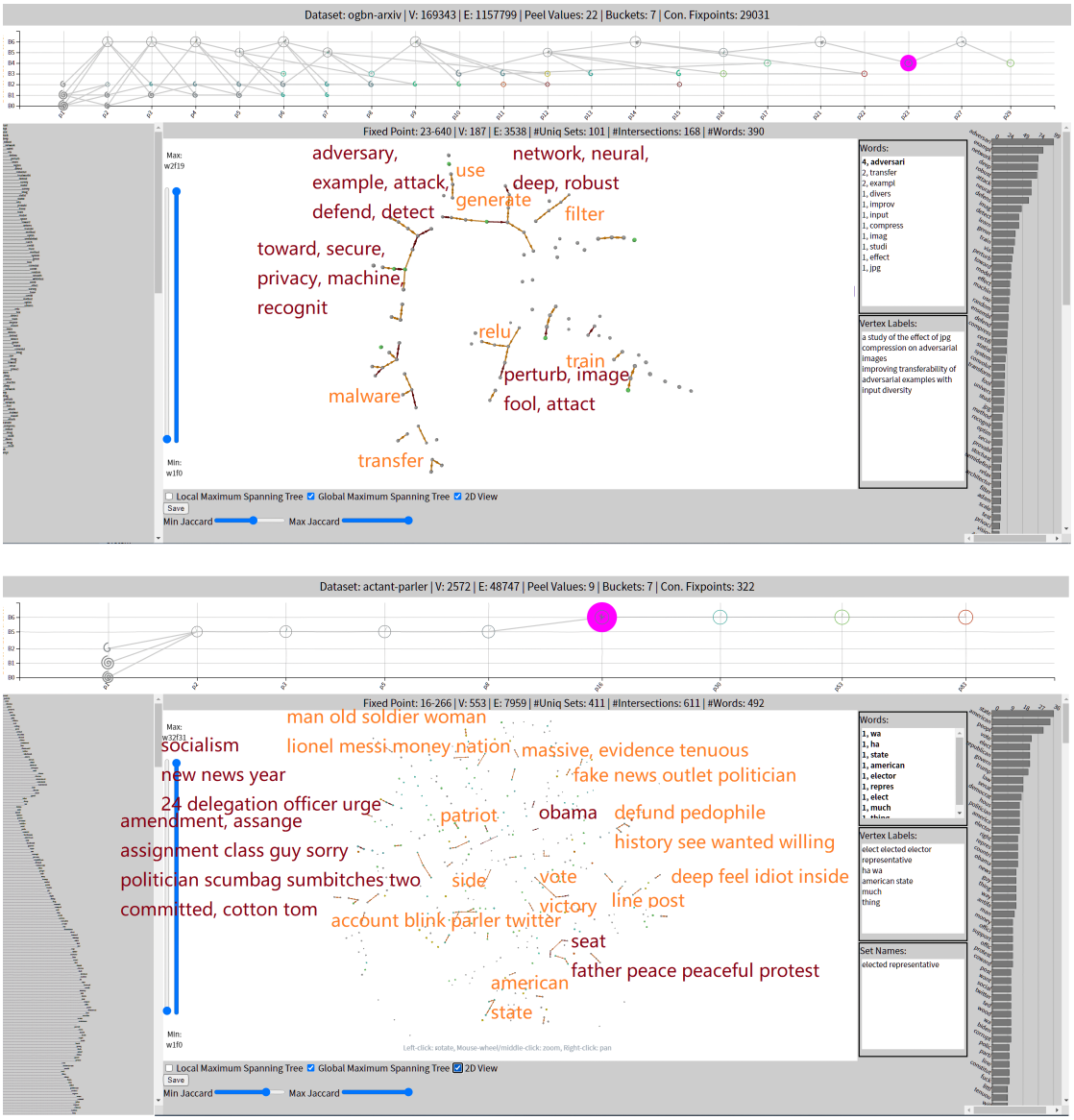

Figure 6: Top figure: Set system intersection meta-graph computed for a fixed point F_{23} from the arXiv Computer Science paper citation network. The displayed labels have a higher (above the mean) co-occurrence frequency in the labels set system. The selected terms have a high level of co-occurrence in the paper titles grouped by F_{23}. Bottom figure: Set system intersection meta-graph computed for a fixed point F_{16} from the Parler dataset. The detailed view of the glyph corresponding to the selected fixed point (in red) is displayed in Figure 3 (middle). The displayed labels have a higher (above the mean plus one standard deviation) co-occurrence frequency in the labels set system. The selected terms have a high level of co-occurrence in the sentences collected in the Parler dataset. Notable selected terms include "Lionel Messi", "Obama", "Assange", "vote", "fake news", "pedophile", "American state", "peaceful protest", and "socialism".

References

[1] J. Abello, D. Nakhimovich, C. Han, M. Aanjaneya, Graph cities: Their buildings, waves, and fragments (2021).

[2] J. Abello, H. Zhang, D. Nakhimovich, C. Han, M. Aanjaneya, Giga graph cities: Their buckets,

buildings, waves, and fragments, IEEE Computer Graphics and Applications 42 (2022) 53–64.
[3] J. Leskovec, A. Krevl, {SNAP Datasets}: {Stanford} large network dataset collection, 2014.
[4] M. M. Aliapoulios, E. Bevensee, J. Blackburn, B. Bradlyn, E. D. Cristofaro, G. Stringhini, S. Zannettou, A large open dataset from the parler social network, in: International Conference on Web and Social Media, 2021.
[5] J. Abello, Hierarchical graph maps, Computers & Graphics 28 (2004) 345–359.
[6] J. Abello, A. L. Buchsbaum, J. R. Westbrook, A functional approach to external graph algorithms, in: European Symp. on Alg., Springer, 1998, pp. 332–343.
[7] A. Aggarwal, S. Vitter, Jeffrey, The input/output complexity of sorting and related problems, Communications of the ACM 31 (1988) 1116–1127.
[8] Z. Lin, N. Cao, H. Tong, F. Wang, U. Kang, Interactive multi-resolution exploration of million node graphs, in: IEEE Conference on Visual Analytics Science and Technology, Poster, 2013.
[9] P. Mi, M. Sun, M. Masiane, Y. Cao, C. North, Interactive graph layout of a million nodes, in: Informatics, volume 3, Multidisciplinary Digital Publishing Institute, 2016, p. 23.
[10] N. Bikakis, J. Liagouris, M. Krommyda, G. Papastefanatos, T. K. Sellis, graphvizdb: A scalable platform for interactive large graph visualization, 2016 IEEE 32nd International Conference on Data Engineering (ICDE) (2016) 1342–1345.
[11] V. Yoghourdjian, T. Dwyer, K. Klein, K. Marriott, M. Wybrow, Graph thumbnails: Identifying and comparing multiple graphs at a glance, IEEE Transactions on Visualization and Computer Graphics 24 (2018) 3081–3095.
[12] K. Van Koevering, A. Benson, J. Kleinberg, Random graphs with prescribed k-core sequences: A new null model for network analysis, in: Proceedings of the Web Conference 2021, 2021, pp. 367–378.
[13] R. Laishram, A. Erdem Sar, T. Eliassi-Rad, A. Pinar, S. Soundarajan, Residual core maximization: An efficient algorithm for maximizing the size of the k-core, in: Proc. of Int. Conf. on Data Mining, SIAM, 2020, pp. 325–333.
[14] N. Wang, D. Yu, H. Jin, C. Qian, X. Xie, Q.-S. Hua, Parallel algorithm for core maintenance in dynamic graphs, in: Prof. of ICDCS, 2017, pp. 2366–2371.
[15] V. Batagelj, M. Zaveršnik, Fast algorithms for determining core groups in social networks, Adv. in Data Anal. and Class. 5 (2011) 129–145.
[16] H. Kabir, K. Madduri, Shared-memory graph truss decomposition, in: 2017 IEEE 24th Int. Conf. on High Perf. Comput. (HiPC), IEEE, 2017, pp. 13–22.
[17] A. Arleo, O.-H. Kwon, K.-L. Ma, Graphray: Distributed pathfinder network scaling, in: 2017 IEEE 7th Symp. on Large Data Anal. and Vis., 2017, pp. 74–83.
[18] W. L. Hamilton, R. Ying, J. Leskovec, Representation learning on graphs: Methods and applications, IEEE Data Engineering Bulletin (2017).
[19] R. Yang, J. Shi, X. Xiao, Y. Yang, S. S. Bhowmick, Homogeneous network embedding for massive graphs via reweighted personalized pagerank, Proceedings of the VLDB Endowment 13 (2019) 670 – 683.
[20] J. Abello, D. Nakhimovich, Graph waves, in: The 3rd International Workshop on Big Data Visual Exploration and Analytics with EDBT/ICDT, 2020.
[21] K. Wang, Z. Shen, C. Huang, C.-H. Wu, Y. Dong, A. Kanakia, Microsoft academic graph: When experts are not enough, Quantitative Science Studies 1 (2020) 396–413.
[22] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Advances in neural information processing systems 26 (2013).
[23] J. Abello, F. Queyroi, Fixed points of graph peeling, in: Proc. of the 2013 IEEE/ACM Int. Conf. on Adv. in Soc. Net. Anal. and Mining, 2013, pp. 256–263.
[24] A. Rajaraman, J. D. Ullman, Mining of massive datasets, 2011.
[25] D. Chandrasekaran, V. Mago, Evolution of semantic similarity—a survey, ACM Computing Surveys (CSUR) 54 (2021) 1–37.