Importance driven focus+context
for inspiration analysis in citation graphs

Marius Irgens

November 2022

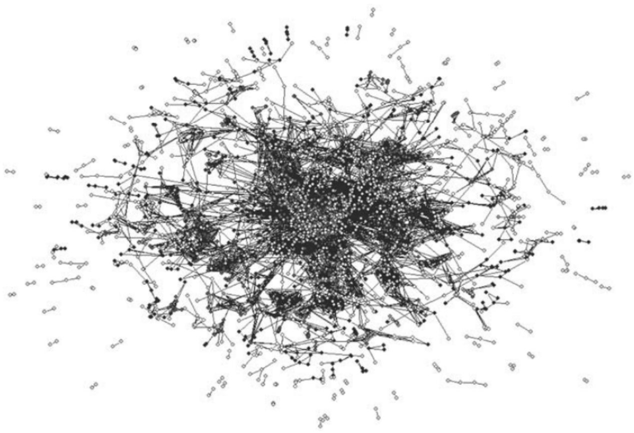
1 Abstract

This paper introduces importance-driven citation graph rendering as a novel technique for focus+context display of inspiration. It uses author-to-author citations as an example: How inspired is a source-author by the other authors, and how inspired are the other authors by that source-author. However, the concept is not exclusive to authors, and could be applied to any set of citation givers and takers; science groups or scientific journals for instance.

2 Introduction

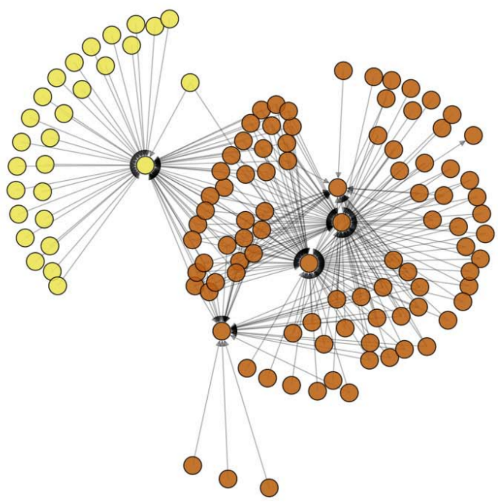
A citation graph is a directed graph that describes the distribution of citations within a set of citation givers and receivers. The citation givers and receivers can be papers, authors, science groups, or even scientific journals.
Citation graphs are often big and dense, with many overlapping nodes and edges. To be able to analyse such a graph, it is important to have the option to visualize it in different ways, depending on what is to be analysed. This calls for interactive functionality. Even in simple usecases, the visualization mantra "Overview first, zoom and filter, then details-on-demand" become important as nodes and edges start to overlap (figure 1 and 2).

If the goal is to visually analyse single elements in such crowded graphs, interactivity becomes even more important. As an example, analysis of one specific element in figure 1 (even just *finding* it!) is virtually impossible without some form of interactive functionality. But if the visualization *was* interactive, what kind of interactive functionality would provide most useful? It depends on exactly what is to be analysed.
As mentioned in the abstract, this paper will focus on analysis of author *inspiration*. To do this, one needs tools for isolating the inspiration of specific authors, as well as communicating its degree within that isolated area. To isolate, I use concepts from the focus+context paradigm. To communicate inspirational degree, I use concepts from the degree-of-interest and importance-driven rendering paradigms.
I will first provide a mathematical description of inspiration in section 2.1, then argue why inspiration visualization is useful in section 2.2. In section 3 I present an overview of the scientific work this paper is based on. In section 4 I try my best to explain the procedure of inspiration visualization, before presenting my results in section 5. A short discussion follows in section 6.



Leydesdorff, 2004, p.379

Figure 1: A very dense journal-to-journal citation graph for analysis of journal clusters [1]. This is a static image. Interactive functionality like zoom, filter or details-on-demand would make cluster analysis much more sophisticated. Zooming with name labels on-demand could give detailed information about specific journals. Color filters could help identify clusters, with cluster attributes interactively definable by the user.



Chandrasekharan, 2021, p.198

Figure 2: A sparse author-to-author citation graph, also for analysis of clusters [2]. Simply adding color filtering helps cognition immensely.

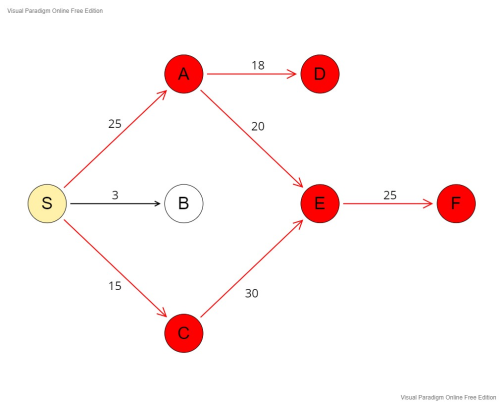


Figure 3: Author S is the source. Author B is only cited 3 times and therefore not counted as inspiring. Author A and C are directly inspiring. Author D and E is indirectly inspiring with distance 2. Author F is indirectly inspiring with distance 3.

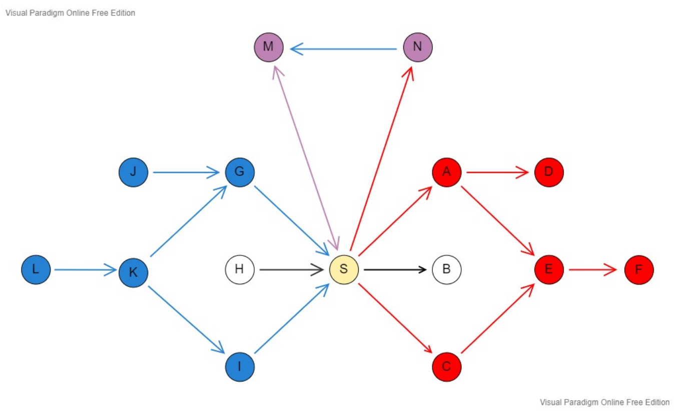


Figure 4: The red nodes represent authors that author S is inspired by. The blue nodes represent authors inspired by author S. The purple nodes represent authors in a reciprocal inspirational relationship: Author S is directly inspired by author M, and author M is directly inspired by author S back. Author S is directly inspired by author N, but author N is only indirectly inspired by author S back. The purple edge represents equidistant reciprocal inspiration.

2.1 What is inspiration?

There are two main categories of inspiration that should be visualized: an authors inspiration *from* other authors, and an authors inspiration *to* other authors. A third category of reciprocal inspiration emerge when authors belong to both. How can one define author-to-author inspiration mathematically? I propose the following:

1. An author is inspired by another author if his total number of citations towards that author exeeds some reasonable minimum threshold. I think a threshold of 8 citations is a good starting point for authors. You might argue this is to much, or to little, and I might agree. This threshold is variable, depends on the situation, and should optimally be user-definable.
2. An author is indirectly inspired by another author if he is inspired by someone who is inspired by that author. This relation is recursive, and the indirect inspiration distance from one author to another equals the least number of inspirational jumps needed to get there (the shortest path).
3. The inspirational degree is further defined by the total number of citations in an inspiration connection. An author is more inspired by someone he cited 50 times than someone he cited 10 times; this should be accounted for.

When a source-author is defined, who he is inspired by can be visualized as a tree (figure 3). Which authors are inspired by the source-author can likewise be visualized as a tree, by using the same logic in the opposite direction. Something very interesting happens when visualizing both at the same time that requires special treatment - cyclic graphs emerge (figure 4).

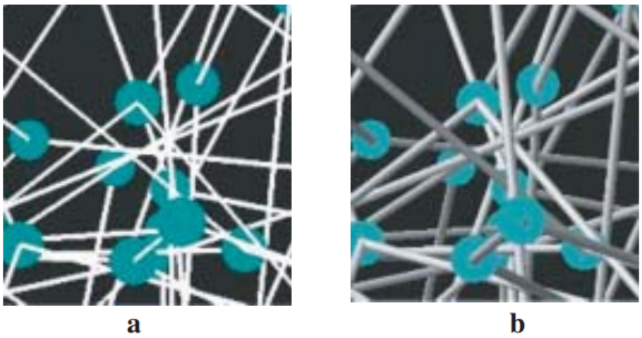
2.2 why is inspiration useful?

Analyzing individual authors inspiration can be very useful when one needs to evaluate them and their scientific work. It can also highlight group dynamics. Sometimes it can reveal relations even the authors themselves did not know about. It can also teach us something about the structure of citation graphs in general. In "Lattices in citation networks: An investigation into the structure of citation graphs" [3] Yong et. al. analyse the structure of paper-to-paper citation graphs, highlighting a handful of lattice substructures that are considered scientifically meaningful - the most important being the simple diamond shape: one paper cites two other papers, that in turn cites the same third paper, showing both co-citations and bibliographic coupling in one simple shape. Individual authors inspiration graphs have different meaning than paper-to-paper citation graphs, but the shapes highlighted in Yongs studies do appear in them as well. The introduction of reciprocal inspiration gives rise to new cyclic structures not yet covered in Yongs studies. These are very interesting, and can surely teach us something new about citation graph structures not covered in previous studies.

3 Related Work

3.1 Focus+Context and importance-driven rendering

To let the user isolate a subset of information and still be able to compare it to the full set it exists in, one can use focus+context. Applying focus+context concepts to a citation graph can help a user understand the inspiration of a selected source-author, compared to the total amount of inspiration in the author set. Each author contains a subgraph consisting of an incoming inspiration tree, an outgoing inspiration tree, and a number of cyclic graphs of reciprocal inspiration. This subgraph can be defined as a selectable focus object. All authors and citation edges *not* included in it is considered context. Separate rendering techniques can be used on focus and context to heighten the information bandwidth. In "Generalizing Focus+Context Visualization", Hauser et. al. mentions color, opacity and style as possible graphical resources to differentiate focus from context [4].
Importance-driven rendering concepts can be applied to give the user better control of the focus reach and the render characteristics within it to heighten the information bandwidth further. In "Degree-of-interest trees: a component of



Ware, 2008, p.8

Figure 5: (a) Line rendering; (b) Tube rendering.

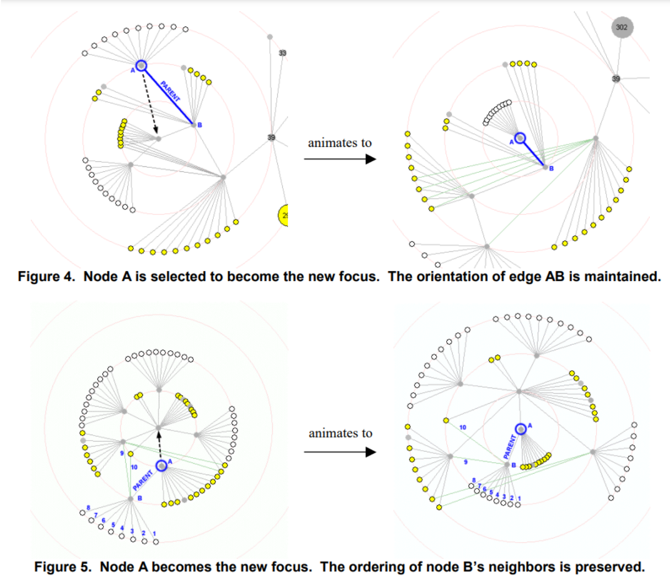
an attention-reactive user interface" [5] by Card et. al, the intrinsic importance of a node (its "degree-of-interest") is defined as its distance from a focus node. The distance is then defined as "the number of nodes that must be traversed by following parent and child links from the node of interest until reaching the subject node." This intrinsic importance can then be used to alter the characteristics of the render, for instance its color, opacity, style, or geometric transformation. The focus reach can be user-adjustable by control of an additional threshold based on intrinsic importance; for instance, if a node has a distance of 5 or more, it is no longer part of the focus.

3.2 3D graph representation and immersive viewing

Citation graphs can get very dense, especially if the minimum citation threshold is low. It will most likely contain a high number of overlapping nodes and crossing edges. In such a case, depth cues can help the user understand the graph. In "Visualizing Graphs in Three Dimensions" [6], Ware et. al shows how 3D shading techniques can help when edges cross (figure 5). In the same paper, Ware et. al lays out more depth cues and how they help, the most important being: Stereoscopic disparities, kinetic depth, perspective and occlusion.
In "Viewing a graph in a virtual reality display is three times as good as a 2D diagram" [7], Ware et.al, argues depth cues can be enhanced even further by Virtual Reality (VR) peripherals; especially stereoscopic disparities and kinetic depth. There are offcourse several drawbacks to making a visualization VR-only. However, implementing VR functionality as an on-demand extension to a desktop application should be beneficial to any 3D graph visualization. Implementing immersive viewing on-demand as an option gives the user all of the benefits and none of the drawbacks, as long as the implementation did not lead to any compromises.

3.3 Distance-driven graph layout

In "A survey of two-dimensional graph layout techniques for information visualisation" [8] by Gibson et. al, several graph layout considerations is discussed. While not all are applicable to 3D graphs, many are: "how the graph is drawn has a significant impact on how the graph is understood". The most important concepts are the perceived relationship from proximity (the gestalt principle), perceived relationship from



Yee, 2001, p.4

Figure 6: The selected source-node moves to the middle of the view, and the other nodes are laid out radially on circles with radius equal to their distance from the source-node.

similarity (similarity in shape or color for instance) and uniform edge lengths (to keep the graph undistorted).
In "Animated Exploration of Graphs with Radial Layout" [9] by Yee et. al, the graph layout changes dynamically, depending on which node is is selected as the source-node. Each other nodes placement depends on its distance from the source-node (figure 6). In "H3: Laying Out Large Directed Graphs in 3D Hyperbolic Space" [10] Munzner et. al use a similar approach to lay out nodes radially in 3D instead.

4 Method

Although my solution is based on 3D graph representation, these concepts should be equally applicable to 2D graphs. The degree-of-inspiration calculation does not rely on spatial information, and neither does the inspiration-driven transfer-function. Distance-driven force layout can just as well be done in 2D. The decision to use 2D versus 3D should lie with the complexity of the graph: if your graph visualization does not benefit from 3D representation, use 2D instead.

4.1 Adjustable degree-of-inspiration

When a source-author is selected, all other authors can get assigned a degree-of-inspiration value based on their intrinsic importance to the source-author. First, check if the target-author can be reached at all. If there is no path from the source-author to the target-author it has zero intrinsic importance, and there is no need for further calculation. Also, define if the target-author is someone who *inspires* the source-author (outgoing citations), someone who *is inspired* by the source-author (incoming citations), or *both*. Then, based on the target-authors path distance from the source-author, and the number of citations in each step between them, calculate a degree-of-inspiration value. I came up with the following equation:

*insp(x) = inspMax(parent(x)) - (distance(x)/distanceK) - (distance(x)/numCit(x)*citK)*

insp(x) is the resulting degree-of-inspiration of a target-author. *Parent(x)* are all authors with distance = (distance(x) - 1) that also lead to the target-author within the focus graph. *inspMax(parent(x))* is the single highest degree-of-inspiration value of all parents of a target-author. *Distance(x)* is the path distance from the source-author to the target-author. *DistanceK* can be adjusted by the user. *NumCit(x)* is the total number of citations leading to the target-author from his parents. *CitK* can be adjusted by the user. Figure 7 shows an example calculation.
This results in a curved falloff. It is possible to make the falloff linear by replacing *distance(x)* with a constant number. Increasing the *distanceK* value makes distance less influential. Increasing the *citK* value makes the number of citations less influential. Increase or lower both to affect the "reach" of inspiration.
K-value adjustment should be interactive, with lowest possible computation cost to keep the framerate high. There is no need to perform the calculation for every author in the whole set. It is enough to perform it for every author in the focus subset. Parents should also be calculated before children, to keep from having to do unneccessary, recursive calculations (in the spirit of memoization). I suggest the following setup:

1. When a source-author is selected, make one list of all authors that can be reached from the source-author by following a shortest path of outgoing citations (reachable nodes), and a list of the edges inbetween (reachable edges). Make one list of all authors that can reach the source-author by following a shortest path of incoming citations (trailing nodes), and a list of the edges inbetween (trailing edges). Make one list of authors that belong to both (cyclic nodes), and a list of the edges inbetween (cyclic edges). Each node-list element should contain distance, a list of parents, and total number of citations from all parents. Cyclic node-list elements is made by combining the information from both the referenced reachable node-list element and the trailing node-list element: concatenate the lists of parents, add together the number of citations, and use the lowest distance value of the two.
2. Sort these lists by distance, ascending.
3.When the user changes the K-values, iterate through these lists in ascending fashion and perform the degree-of-inspiration calculations. Start with reachable and trailing nodes, then cyclic nodes. This will make sure all parents are calculated first.

4.2 Inspiration-driven transfer-function

To visualize an authors degree-of-inspiration, his representing node needs to be rendered in a manner that reflects it. A transfer function takes the authors inspiration-type and degree-of-inspiration as arguments and returns a render setting. The inspiration-type says which list he belongs to: reachable nodes, trailing nodes or cyclic nodes - these should be rendered differently. The degree-of-inspiration can for instance be used to interpolate between material colors: red to grey if reachable, blue to grey if trailing, purple to grey if cyclic. It could also be used to interpolate between other material traits, or even affine transformation settings like scale.

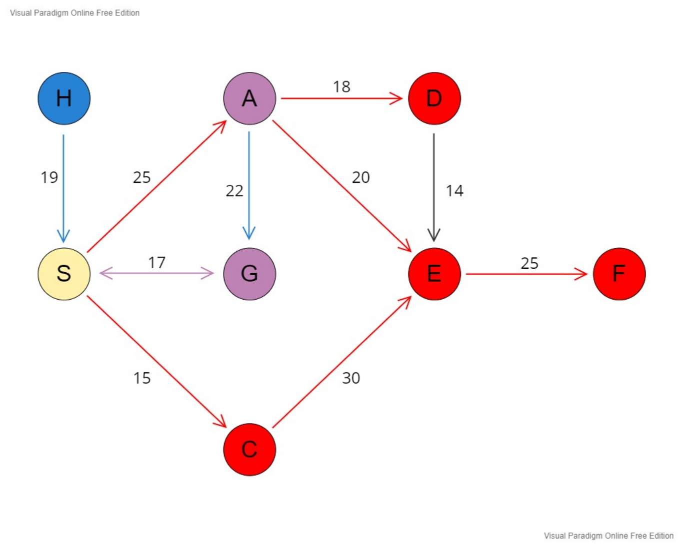


Figure 7: Author S has degree-of-inspiration (DOI) = 1. The user has chosen distanceK = 10 and citK = 1.
Author A has distance = 1 and numCit = 25. His resulting DOI = 1 - 1/10 - 1/25 = 0.86. Author C gets DOI = 0.83.
Author E has distance = 2 and numCit = 50. His resulting DOI = 0.86 - 2/10 - 2/50 = 0.62. Author D gets DOI = 0.55.
Author F has distance = 3 and numCit = 25. His resulting DOI = 0.62 - 3/10 - 3/25 = 0.2.
The edge DE is disregarded since author E has distance = 2, and this edge has distance = 3.
Author G has distance = 1 and numCit = 39. His resulting DOI = 1 - 1/10 - 1/39 = 0.87.
Author H ha distance = 1 and numCit = 19. His resulting DOI = 1 - 1/10 - 1/19 = 0.85.

The edges inbetween can use the degree-of-inspiration value of the node they "belong to" (meaning the author that uses its citation value) for their transfer function. Their directionality should also be expressed clearly: reachable edges, trailing edges and cyclic edges should be rendered differently.
All nodes and edges that does not belong to the focus, and therefore is part of the context, can use a context render setting. In addition, any node or edge within the focus that gets a degree-of-inspiration value below some chosen threshold can be assigned that same context render setting. In this way, the user can adjust the reach of the focus as well as its render qualities by changing the K-values (figure 8).
Cycles can get created or broken by updating the K-values. The transfer function needs to perform a check for every cyclic node to see if it really is part of a cycle, with respect to the current K-values. This can be solved with a recursive function over all parents of the cyclic node, that returns true when the source has been reached, or false otherwise. This might sound computational heavy, but unless the focus graph is very heavily populated with cyclic nodes it should not affect performance much.

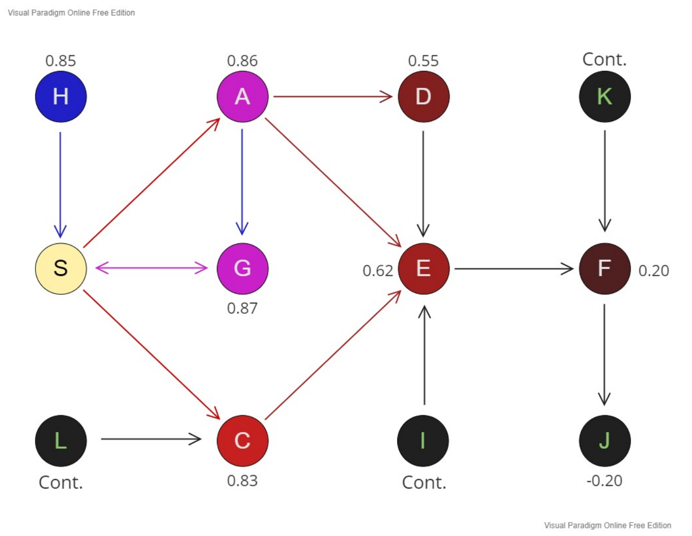


Figure 8: A transfer-function takes inspiration-type and DOI value as argument and returns a color value. Author I, K and L is not part of author S' focus, and is assigned a context material. Author J's DOI-value falls below threshold = 0, and is also assigned a context material.

4.3 Distance-driven force Layout

When the graph is constructed it is assigned a default layout. A static layout can make it hard to trace inspiration paths in a focus. A dynamic layout that can update with respect to the selected source-author, ordered in a parent-child hierarchy, is helpful. A popular dynamic layout concept is force-directed layouts. Nodes are moved by simulating forces on them. A good force simulation supports both animated movement and interactive adjustments of force attributes. For my solution I opted for the popular force simulation API d3-force-3d [11]. It "implements a velocity Verlet numerical integrator for simulating physical forces on particles." It is highly optimized and supports definition of custom forces - which I needed. D3-force-3d is a 3D extension of the 2D force simulation d3-force, which is part of the popular API D3.js. This is how I use it to obtain a distance-driven force layout:

The default graph layout is made using forces that are provided with the API - center force, link length force and collision force. The center force makes sure every node gravitates towards a defined center. The link length force uses spring-force simulation to try to maintain equal edge lengths between connected nodes. The collision force preserves a minimum distance between all nodes so they do not overlap (figure 9).

In addition to these forces, I customized a radial force and a companion straightening force. These are only available when a source-author has been selected, and only apply to nodes in focus. Nodes in the context use the default forces and "follows along", trying to uphold their default force constraints.

Radial force (figure 12): Move the source-author to the center. Every other author gravitates towards or away from the center such that its radial length from the center equals its distance * link length.

Straightening force (figure 13): This force is meant to be added on top of the radial force to make parent-child relationships clearer. For every author with distance 2 or more, try to obtain the same vector angle as the parent.

5 Evaluation

For my evaluation I used javascript ES6 with the following API's: three [12], jsnetworkx [13], d3-force-3d [11], dat.gui [14] and three-mesh-ui [15]. The dataset is derived from the visualization publications dataset [16] and contains author-to-author citations with a minimum citation threshold of 8.

I will start with pictures of the main functionality: dynamic force layout, adjustable K-values and the transfer-function. Then I will show how this can be used to evaluate inspiration of individual authors. Afterwards I show its usage in group dynamics analysis. Finally, I show how it can be used for citation graph structure analysis.

I want to point out that the 3D graph is much easier to read when you can interact with it, rather than looking at still pictures of it. I also added VR on-demand, and viewing the graph immersively really heightens the fidelity, even when using a relatively low-budget HMD like the Meta Quest 2.

5.1 Main functionality

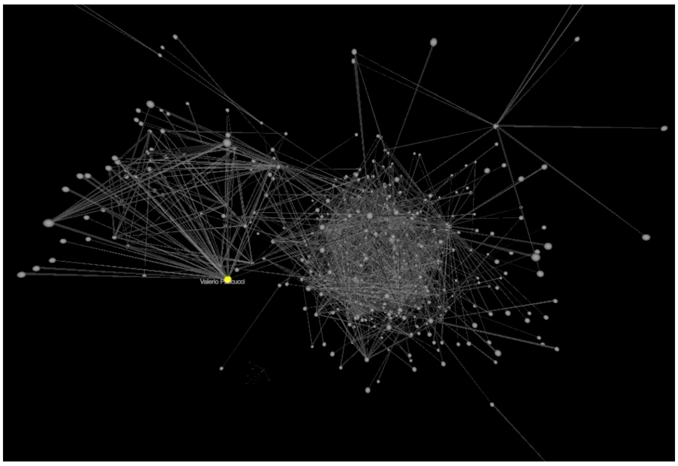


Figure 9: The default layout is arranged by simulating forces on the nodes: center force, link length force and collision force.

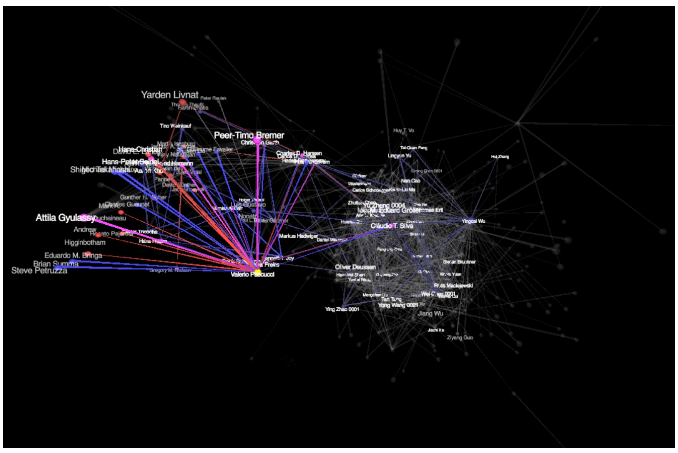


Figure 10: Selecting an author constructs an opaque, colored focus with name labels. The context is rendered semi-transparently.

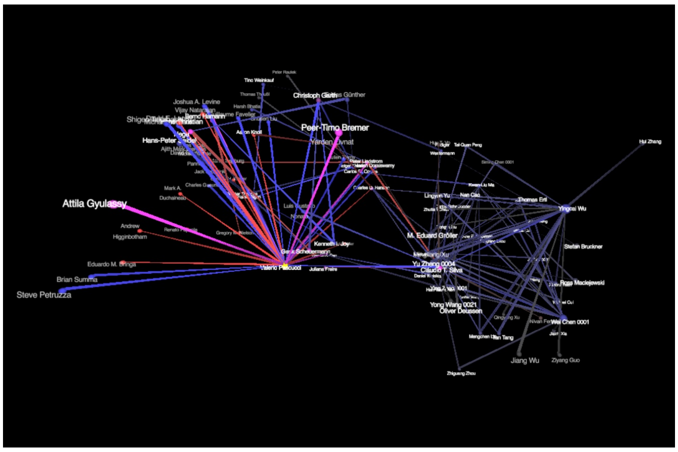


Figure 11: By adjusting the opacity of the context, the user can choose when and how to view it. In this figure (and succeeding figures) the opacity has been turned to 0, effectively removing it altogether.

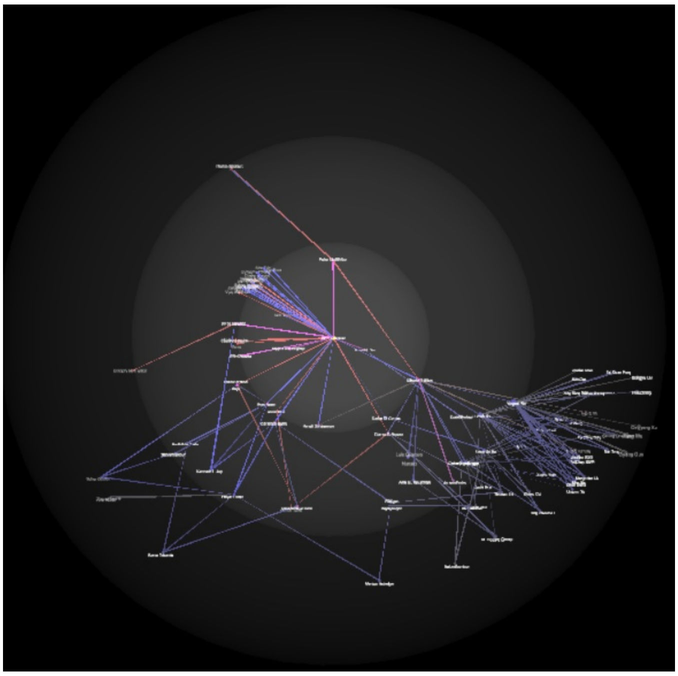


Figure 12: The radial force dynamically rearranges the nodes to easen path-tracing. Semi-transparent spheres help convey radial distance from the center. Parent-child relationships can still be a bit unclear due to angle differences.

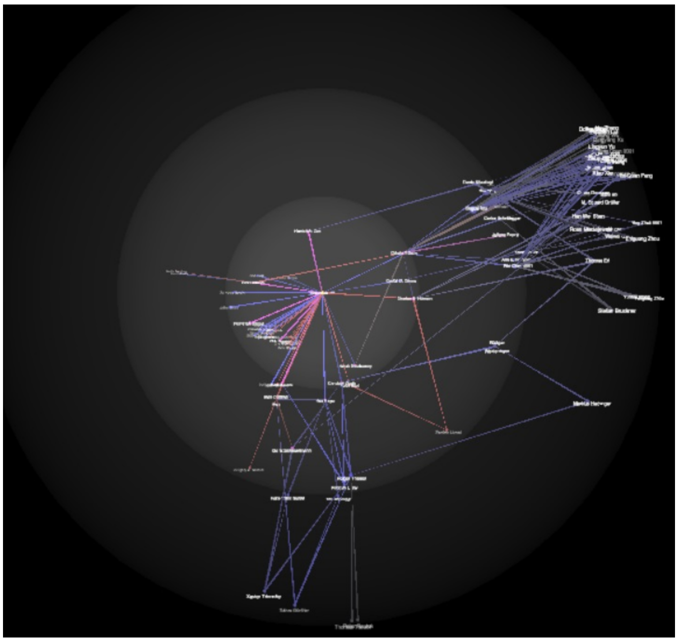


Figure 13: The straightening force makes parent-child relationships clearer.

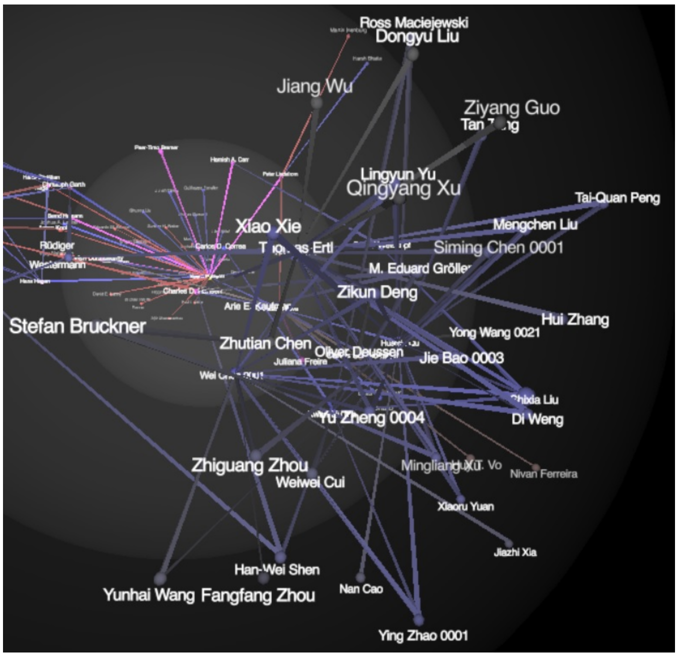


Figure 14: Gaining the minimum distance variable of the collision force separate the nodes to make the names easier to read.

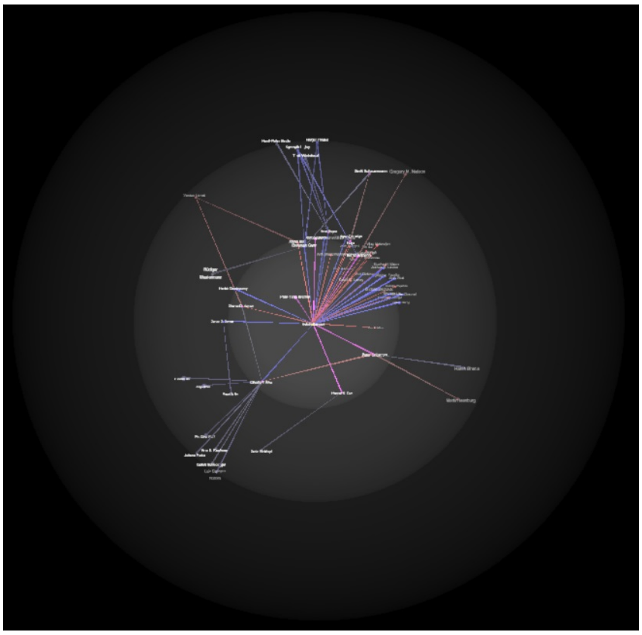


Figure 15: Adjusting the K-values shorten or extend the focus reach. The radial layout can be retriggered to rearrange the graph with respect to the current focus reach. In this figure, distanceK = 12 and citK = 0.8.

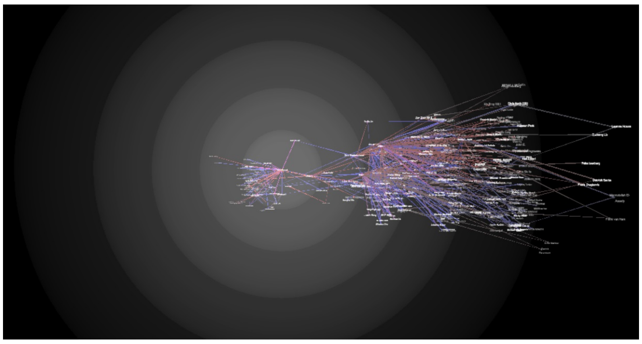


Figure 16: Gaining the K-values can reveal a quite complex graph. In this figure, with distanceK = 50 and citK = 4, the leaves reaches a distance of 6. The number of cyclic graphs within the focus rise drastically, and causes the graph to tangle.

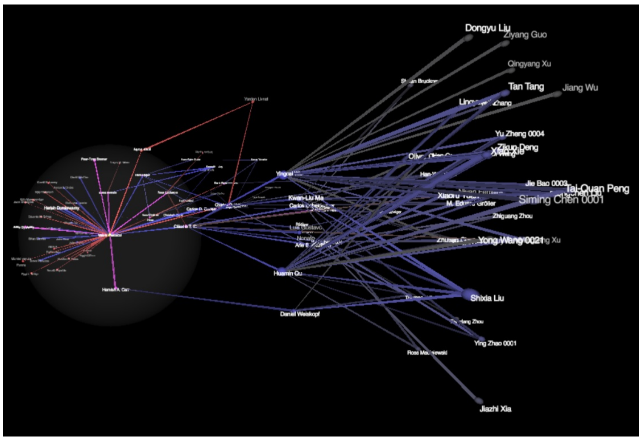


Figure 17: The color of the nodes and edges reflect their degree-of-inspiration values, here interpolating from blue to grey.

5.2 Author inspiration analysis

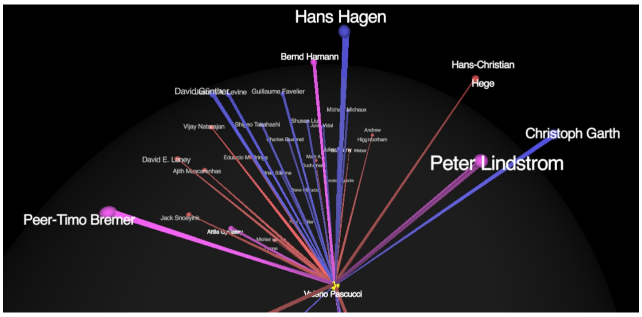


Figure 18: The yellow node is the source-author. Red nodes and edges represents authors that inspired the source-author. Blue nodes and edges represents authors that is inspired by the source-author. Purplpe nodes and edges represents reciprocal inspiration.

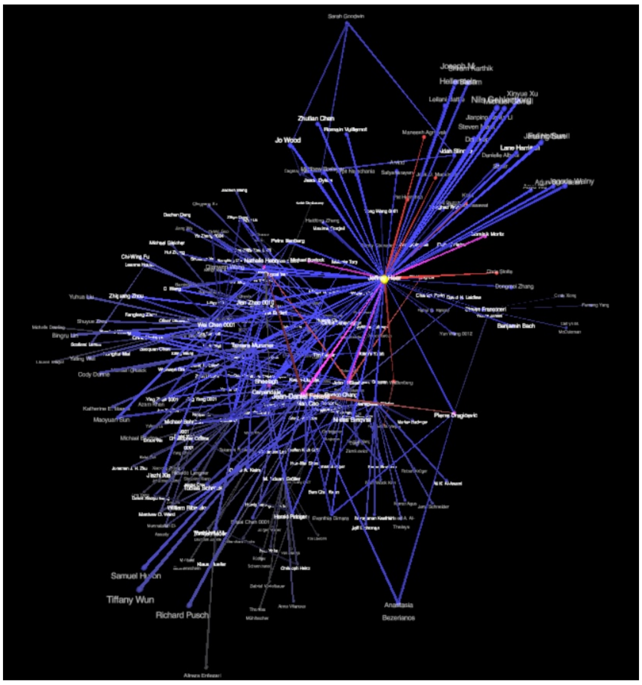


Figure 19: The inspiration graph of Jeffrey Heer: clearly a highly inspirational individual. Not symmetrically inspired back.

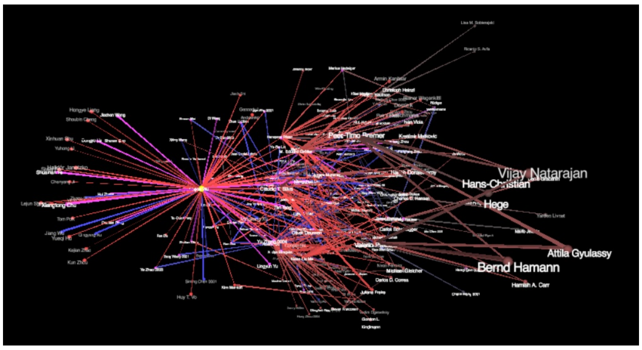


Figure 20: The inspiration graph of Yingcai Wu: almost the opposite of Jeffrey Heer. Very inspired by the visualization community, with a high number of direct reciprocal inspiration as well.

5.3 Group dynamics analysis

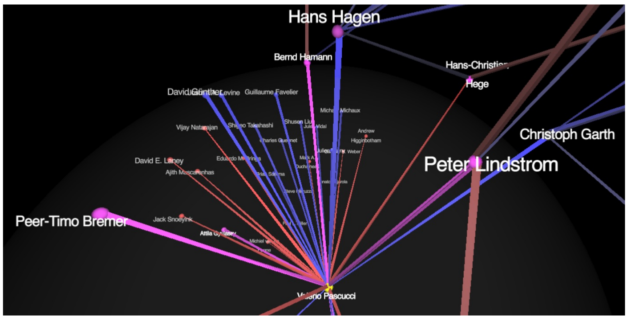


Figure 21: Purple nodes without purple edges can reveal non-equidistant reciprocal inspiration. In this figure, Valerio Pascucci (bottom) is directly inspired by Hans-Christian Hege (top right), but Hans-Christian Hege is only indirectly inspired by Valerio Pascucci through Hans Hagen (top). Also, Hans Hagen is directly inspired by Valerio Pascucci, but Valerio Pascucci is only indirectly inspired by Hans Hagen through Hans-Christian Hege. This circle of inspiration is quite fascinating to me: they are all inspired by eachother, but at different qualities.

Figure 22: Purple nodes without purple edges can also mean equidistant reciprocal inspiration taking different paths. Here, Valerio Pascucci (bottom left) is indirectly inspired by Gerik Scheuermann (right) through Hans-Christian Hege (top), and Gerik Scheuermann is indirectly inspired by Valerio Pascucci through Christoph Garth (middle).

5.4 Structure analysis

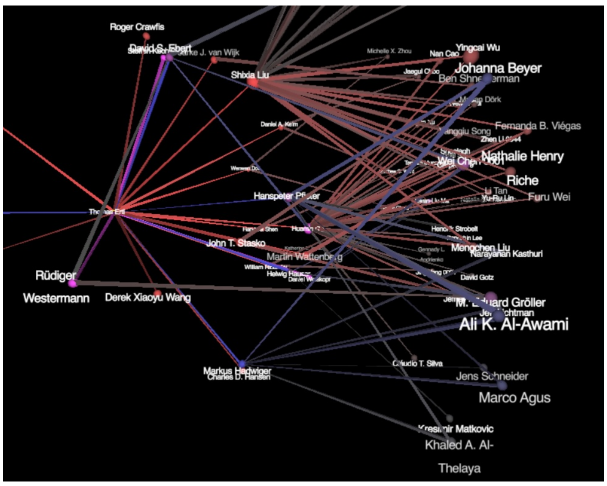


Figure 23: The diamond shape is considered one of the most important substructures in citation graphs [3]. Thomas Ertls inspiration graph contains an incredible amount of them, grouped together in what I call "multidiamond groups".

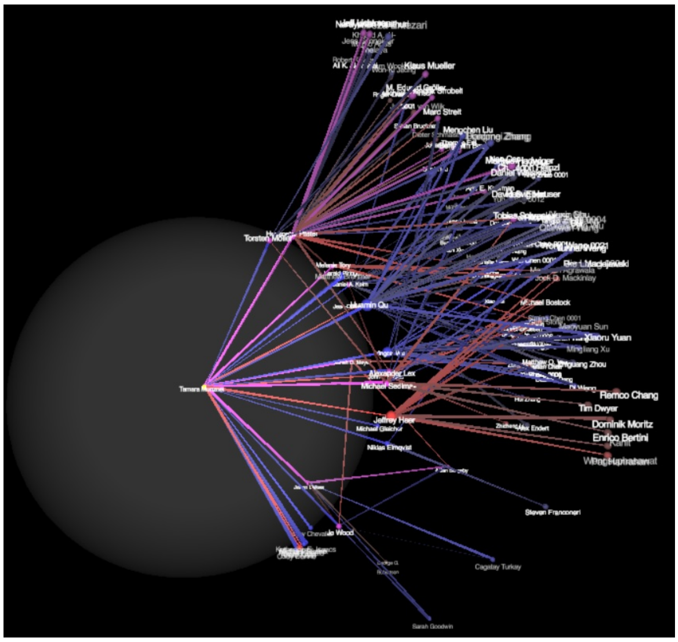


Figure 24: Tamara Munzners inspiration graph has a high amount of what I call "sprays": connections with distance = x, that diverge into a very high number of connections with distance = (x + 1). They are at distance = 1, which means she is directly inspired by highly inspired authors, and directly inspirational for highly inspirational authors.

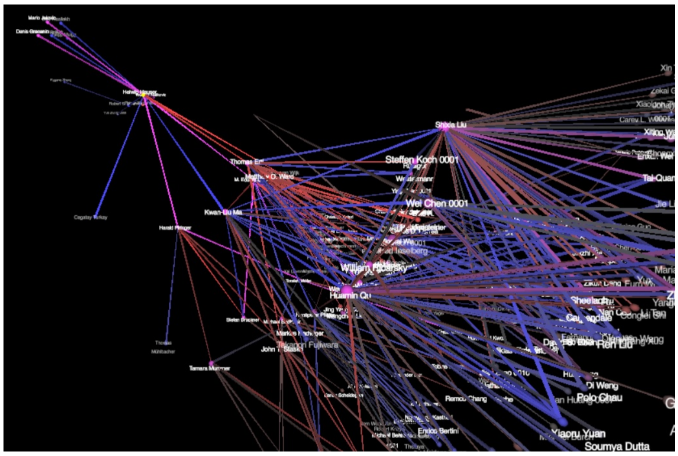


Figure 25: Helwig Hausers inspiration graph contains a high amount of "spray" cross-connectivity, causing a tangled mess of reciprocal inspiration. It has a high degree of "spray entanglement".

6 Conclusion

Importance driven focus+context for inspiration analysis can be implemented as part of a citation graph application. It can be used to learn about individual authors, as shown in section 5.2. Relatively complex group dynamics can also be revealed, as shown in section 5.3. Section 5.4 shows how citation graph structures can be studied from a new angle, by using the concept of inspiration to build new subgraphs that can be analysed and compared individually.

6.1 Applications

As mentioned, these ideas are not restricted to author-to-author citation sets; but they will need some tweaking to fit other types of sets. For instance, a journal-to-journal citation set probably has a very high number of citations between journals. A well-suited minimum citation threshold should be used to define journal-to-journal inspiration, and the K-value input range set such that users achieve a satisfying span of focus reach.

These ideas could be used for other datasets as well; twitter retweet graphs for instance. Instead of author-to-author citations, one could use profile-to-profile retweets. Retweets are basically citations. Instead of number of citations, one could use number of retweets, or even the total number of clicks generated by the retweets. This could create a graph with Twitter profiles as nodes, and edges showing how much traffic profiles generate for eachother. Indirect traffic means one Twitter profile directs traffic towards a second Twitter profile, that in turn directs traffic towards a third Twitter profile; in effect, the first profile indirectly generating traffic towards the third profile, either knowingly or unknowingly. This kind of traffic distribution analysis could be useful in a range of applications.

Another usecase could be evaluating peoples connective status by adding up their "amount of connectivity" through some connective apparatus - for instance phones: instead of number of citations, add up the time people have spent speaking together over phone. The directionality depends on the call instigator. Now one can evaluate to which degree people are indirectly connected, by looking at their phonecalls. It would also allow for structural analysis of phonecall graphs.

References

[1] Loet Leydesdorff. "Clusters and maps of science journals based on bi-connected graphs in journal citation reports". In: *Journal of documentation* (2004), 371–427 . DOI: https://doi.org/10.1108/00220410410548144.

[2] Shreya Chandrasekharan et al. "Finding scientific communities in citation graphs: Articles and authors". In: *Quantitative Science Studies* 2.1 (2021), pp. 184–203. DOI: https://doi.org/10.1162/qss_a_00095.

[3] Fang Yong and Ronald Rousseau. "Lattices in citation networks: An investigation into the structure of citation graphs". In: *Scientometrics* 50.2 (2001), pp. 273–287. DOI: https://doi.org/10.1023/A:1010573723540.

[4] Helwig Hauser. "Generalizing focus+ context visualization". In: *Scientific visualization: The visual extraction of knowledge from data*. Springer, 2006, pp. 305–327. DOI: doi.org/10.1007/3-540-30790-7_18.

[5] Stuart K Card and David Nation. "Degree-of-interest trees: A component of an attention-reactive user interface". In: *Proceedings of the Working Conference on Advanced Visual Interfaces*. 2002, pp. 231–245. DOI: doi.org/10.1145/1556262.1556300.

[6] Colin Ware and Peter Mitchell. "Visualizing graphs in three dimensions". In: *ACM Transactions on Applied Perception (TAP)* 5.1 (2008), pp. 1–15. DOI: doi.org/10.1145/1279640.1279642.

[7] Colin Ware and Glenn Franck. "Viewing a graph in a virtual reality display is three times as good as a 2D diagram". In: *Proceedings of 1994 IEEE Symposium on Visual Languages*. IEEE. 1994, pp. 182–183. DOI: doi.org/10.1109/VL.1994.363621.

[8] Helen Gibson, Joe Faith, and Paul Vickers. "A survey of two-dimensional graph layout techniques for information visualisation". In: *Information visualization* 12.3-4 (2013), pp. 324–357. DOI: doi.org/10.1177/1473871612455749.

[9] Ka-Ping Yee et al. "Animated exploration of graphs with radial layout". In: *Proc. IEEE InfoVis 2001*. 2001, pp. 43–50. DOI: doi.org/10.1109/INFVIS.2001.963279.

[10] Tamara Munzner. "H3: Laying out large directed graphs in 3D hyperbolic space". In: *Proceedings of VIZ'97: Visualization Conference, Information Visualization Symposium and Parallel Rendering Symposium*. IEEE. 1997, pp. 2–10. DOI: 10.1109/INFVIS.1997.636718.

[11] *d3-force-3d: force simulation API (github repository with documentation)*. https://github.com/vasturiano/d3-force-3d. Accessed: 2022-11-11.

[12] *three: 3D graphics API (homepage with documentation)*. https://threejs.org. Accessed: 2022-11-24.

[13] *jsnetworkx: network/graph API (homepage with documentation)*. https://felix-kling.de/jsnetworkx. Accessed: 2022-11-24.

[14] *dat.gui: javascript GUI API (github repository with documentation)*. https://github.com/dataarts/dat.gui/blob/master/API.md. Accessed: 2022-11-24.

[15] *three-mesh-ui: Text labeling and VR UI API (github repository with documentation)*. https://github.com/felixmariotto/three-mesh-ui. Accessed: 2022-11-24.

[16] *Vispubdata: a dataset of visualization publications (website with download link and details about the dataset)*. https://sites.google.com/site/vispubdata/home?authuser=0. Accessed: 2022-11-24.