Flowmapper.org: a web-based framework for designing origin–destination flow maps

Caglar Koylu*, Geng Tian and Mary Windsor

Geographical and Sustainability Sciences, University of Iowa, Iowa City, IA, USA

ABSTRACT
FlowMapper.org is a web-based framework for automated production and design of origin-destination flow maps. FlowMapper has four major features that contribute to the advancement of existing flow mapping systems. First, users can upload and process their own data to design and share customized flow maps. The ability to save data, cartographic design and map elements in a project file allows users to easily share their data and/or cartographic design with others. Second, users can generate customized flow symbols to support different flow map reading tasks such as comparing flow magnitudes and directions and identifying flow and location clusters that are strongly connected with each other. Third, FlowMapper supports supplementary layers such as node symbols, choropleth, and base maps to contextualize flow patterns with location references and characteristics. Finally, the web-based architecture of FlowMapper supports server-side computational capabilities to process and normalize large flow data and reveal natural patterns of flows.

1. Introduction

Flow maps illustrate movements of tangible and intangible phenomena between locations. While there are many types of flow maps, origin–destination (OD) flow maps illustrate directed flows between origin and destination locations while ignoring the actual routes between locations (Dent et al., 2008; Slocum et al., 2009; Tobler, 1987). There have been recent advancements in web-based flow mapping applications. Stephen and Jenny (2017) designed a flow map application that allows users to obtain an overview of flow patterns in U.S. domestic migration. State-to-state migration flows are visualized using Jenny et al.’s (2017) force-directed layout. The application supports details-on-demand functionality by allowing users to select and visualize county-level flows between the selected state and other states using a circular layout. Nost et al. (2017) developed HazMatMapper to explore potential environmental justice issues by visualizing hazardous waste flow data. Using HazMatMapper, users can acquire an overview first then filter the dataset in greater detail with a regional view and an information panel that provides additional statistics. Flowmap.blue is a web-based flow mapping application powered by WebGL (Parisi, 2012) that allows rendering, interactive filtering and animation of a large number of flow lines and nodes using GPU computation (Boyandin, 2021). Its interface provides a variety of base maps and straight flow lines with half-arrows at the end points to depict flow direction. Alternative flow line symbology, such as curved flow lines, is currently not available in Flowmap.blue. Flowmap.blue includes a location clustering feature to aggregate flows between clusters that are formed by nearby units. Location clustering reduces visual cluttering of flows and allows visualization of flow patterns at different scales with zooming. However, the clustering method groups nearby locations while disregarding flow connections. Kepler.gl is another WebGL-powered platform that can render millions of points and thousands of arc lines to produce flow maps for large data (He, 2018). Kepler.gl can perform spatial aggregations on the fly and visualize spatial features such as points, paths, polygons, grids and hexbins, as well. Despite its scalability to visualize large number of features, Kepler.gl offers a limited number of symbology and scaling choices. Also, flow lines are monotone curved arcs, and there is little control over basic cartographic elements and principles such as flow line style, contextual layers, base maps, and references, map elements, etc. Both Flowmap.blue and Kepler.gl include temporal filtering to generate animations for temporal flow data and geographic filtering to highlight flow lines (arc) from and to selected origin and destination locations. Despite these new developments there is still much work to be done to develop open-source flow mapping systems that can process and normalize flow data with computational capabilities.
capabilities, and design and share customized flow maps in professional cartographic workflow with greater customizability and design flexibility.

In this article, we introduce a web-based framework, https://flowmapper.org, to design and share OD flow maps in an interactive environment. Our goal is to provide users greater customizability and design flexibility that could help produce an interactive visualization and a professional static map product that requires minimal touch-up in other graphics software. FlowMapper has four major contributions to web-based flow mapping. First, FlowMapper allows users to upload and process their own data to produce, customize and share flow maps. The ability to store data, cartographic design and map elements in a project file allows users to easily share their data and/or styling with others. Second, users can customize the flow line symbology by including options to change the flow line style, width, and coloring. FlowMapper includes algorithms for drawing curved line styles with varying thickness along a flow line, which reduces the visual cluttering and overlapping by tapering flow lines at origin and destination points. The ability to customize flow symbology supports different flow map-reading tasks such as comparing flow magnitudes and directions and identifying flow and location clusters that are strongly connected with each other. Third, FlowMapper supports supplementary layers such as node symbol, choropleth, and base maps to contextualize flow patterns with location references and characteristics such as net-flow, gross flow, net-flow ratio, or a local attribute such as population density and income. FlowMapper supports user interactions to zoom, filter, and obtain details-on-demand functions to support visual information seeking about nodes, flows and regions. Finally, the web-based architecture of FlowMapper supports server-side computational capabilities to process, normalize and summarize large flow data and reveal natural patterns of flows.

2. Related work

There are several remaining challenges for developing effective flow mapping applications. First, flow map comprehension is challenging due to the visual cluttering of flows even in small data sets. For example, interstate migration within the U.S. between 2015 and 2020 form a network of state-to-state migration flows, which consists of 50 nodes (states), thousands of links and millions of migrants moving through those links between the states. Displaying all flows makes it impossible for the map reader to identify flow patterns. Filtering is used to address visual cluttering by displaying a subset of flows that exceed a threshold value, or the top rank flows, or flows to and from a selected location (Tobler, 1987). However, filtering lacks the ability to provide an overview of flow patterns. Edge bundling is commonly applied to bundle flow lines to reduce cluttering and improve clarity (Graser et al., 2019; Holten & van Wijk, 2009; Phan et al., 2005). However, bundled edges may be perceived as actual routes of flows, which also makes it difficult for the map reader to follow the connections between origins and destinations (von Landesberger et al., 2016). To avoid visual cluttering of flows, OD data can be visualized as a matrix where rows represent the locations of flow origins and columns the locations of destinations (Ghoniem et al., 2005). Wood et al. (2010) introduced OD map based on a matrix visualization that preserves the spatial layout of all origin and destination locations and reveals spatial associations between pairs of locations. Yang et al. (2017) expanded the OD map (Wood et al., 2010) by integrating geographic embedding and cartographic interactions including highlighting, filtering and zooming. However, complex connections between locations are hard to identify in matrix and the OD map visualizations.

Flow maps convey different types of information such as flow magnitude, directions, clustering, and spatial focusing, which requires different symbology choices to best communicate the type of pattern to users. Despite the increased interface complexity (Vincent et al., 2019), users must be provided alternative symbology choices to support different types of flow map-reading tasks (Koylu & Guo, 2017). As compared to straight flow lines, curved flow lines have been shown to enhance readability in graph visualization (Holten et al., 2011) and flow maps (Jenny et al., 2018; Koylu, 2014). A flow map eye-tracking study (Dong et al., 2018) revealed that users could more accurately interpret curved flows over straight flows. Dong et al. (2018) also found that the use of a color gradient is more effective for the perception of flow magnitude than the use of line thickness. Jenny et al. (2018) outlined design principles for curving of flows in origin−destination flow maps. Major principles include minimizing flow line intersections, obtaining large angles at intersections, drawing symmetric single and gradual curves, avoiding flows to pass through nodes and narrow angles between flows at shared nodes. Jenny et al. (2017) adopted a force-directed graph drawing algorithm to follow these principles and create OD flow maps that depict a one-way connection (net-flow) between two locations. The force-directed algorithm used by Jenny et al. (2017) is less effective for reducing visual cluttering in two-way directed flow maps in which there are two flow connections between a pair of locations. This is because the graph drawing algorithm cannot effectively support the design principles when there are too many flow lines and especially two flow lines between the same pair of nodes. In addition,
although symmetric curves may enhance flow map readability for identifying flow connections, magnitudes, and clustering, other user study findings suggest that asymmetric curvature enhances the perception of flow directions (Koylu & Guo, 2017; Ware et al., 2014). Therefore, the best use of flow symbolization may depend on the data and scale because both affect how a map reader scans the map and which features become visually salient.

Computational methods such as flow-based regionalization (Guo, 2009), location-based clustering (Andrienko et al., 2017; von Landesberger et al., 2016), and flow data smoothing and clustering (Guo & Zhu, 2014; Tao & Thill, 2016; Zhu et al., 2019) are used to summarize flows and reveal hidden patterns in flow data. Clustering and regionalization methods aggregate individual flows into flows between clusters of locations or contiguous regions and reduce the number of regions and flows to be displayed. Flow clustering and density estimation methods also reduce the number of flows by identifying and visualizing clusters of flows rather than displaying all flows. Flow clustering methods show great promise for summarizing large flow data sets, providing an overview of patterns with multi-scale mapping ability. However, these methods are computationally expensive and may yet be provided as asynchronous processing tools on a web-based flow mapping environment. In addition to summarizing flows in geographic space, researchers developed methods to visualize temporal flow trends (Andrienko et al., 2017; Boyandin et al., 2011; von Landesberger et al., 2016). Main challenge in time-variant flow mapping is the large number of spatial units and the perceptual challenge associated with detecting changes in flows between time periods. To reduce this perceptual challenge, Boyandin et al. (2011) implemented a heat map to visualize the changes in flow magnitudes or node measures such as inflows and outflows between two time periods. Alternatively, von Landesberger et al. (2016) used the similarity of flow structures to cluster and decrease the number of time periods to be compared. Similarly, Andrienko et al. (2017) clustered time periods based on the similarity of flows and used glyphs to visualize direction and distance patterns of flows. Nevertheless, there is still much work to be done for designing and evaluating time-series flow maps, which we plan to address in our future work.

3. Map design

FlowMapper consists of a flow map with alternative flow line designs, a node (point) symbol map, a choropleth map, and a reference base map. Our goal is to help users design and share their customized flow maps and also export their maps such that they can be polished and finalized with minimal touch-up in other graphics software.

3.1. Flow map

FlowMapper allows users to customize the flow line symbology by including options to change the flow line style, width (thickness), and coloring.

3.1.1. Flow line style

We introduce algorithms for drawing straight and curved flow paths. Unlike monotone line thickness along the flow line, we reduce the overlapping of flow lines at the origin and destination locations by tapering the flow lines to thin points at the extremities. FlowMapper currently provides four flow line designs: Curve with half-arrow, tapered curve, tear-drop curve, and straight line with half-arrow (Figure 1). Curved paths are asymmetric to enhance readability of directions in two-way flow maps (Koylu & Guo, 2017). All flow paths are connected to the outer edge of the circles at each flow’s start and end points instead of the center of circles. The algorithms and directions for producing the cubic Bezier flow symbols are explained in detail in Appendices Section 1.

3.1.2. Flow line (path) thickness

Users can choose a proportional scale to make flow thickness proportional to flow magnitude. Perceptual issues related to proportional symbol mapping are also inherent in proportional flow thickness: It is difficult to perceive differences in the thickness of flows that are scaled proportionally. Moreover, longer flows are visually more salient than shorter flows, which is a similar problem to the perception of large areas in choropleth maps. Also, there are only a few large flows and many small flows in most flow data sets, which makes it challenging for comparing flow magnitudes. Similar to graduated (or range-graded) point symbols, we provide classification methods such as natural breaks, equal interval, quantile, and manual classification for classifying flow magnitudes.

3.1.3. Flow line color

Users can select a single color or a color gradient to fill the flow line path resulting in a double visual encoding (flow color value and thickness) to visualize flow magnitude. For the color gradient, users can either select a classification method with a color scheme adopted from colorbrewer.org or use an unclassed (continuous) color scheme using min–max scaling of flow magnitudes between two selected colors. Users may use a flow stroke, which may enhance the perception of flow lines on parts of a map that multiple flow lines cross over. Figure 2 illustrates proportional symbol legends with a single
color (Figure 2(a)) and a continuous color scheme from white to blue with a gray flow stroke (Figure 2(b)). Proportional symbol legends include three values: minimum, average, and maximum. Figure 2(c) illustrates a classified flow thickness and color with five classes (light to dark green) using a quantile classification.

3.2. Node symbol map

Nodes are the coordinates of origin and destination locations for flows. While a node data set with unique identifiers and coordinates is required to produce a flow map, users may choose to hide node symbols by using zero stroke-width and full transparency for fill color. The choice of whether to use node symbols depends on the data set and task. For example, node symbols are good anchors for airport locations for visualizing passenger flows from and to airports. On the other hand, hiding node symbols for visualizing flows between regions such as states may be preferred to highlight the nature of region-to-region flows.

3.3. Choropleth map

A choropleth map is an optional map for contextualizing flow patterns with regional flow or attribute characteristics. Choropleth maps are especially important when flow data is from regions-to-regions such as state-to-state migration flows. It is beneficial to use a choropleth map to illustrate a location-based measure of flows such as net-flow ratio, migration efficiency, or a locational attribute such as population density.

3.4. Base map, map projections, and map elements

FlowMapper supports base maps produced by ESRI, Stamen and OpenStreetMap. For ESRI base maps, users may choose to add reference labels for places. For regional maps, FlowMapper currently supports the Albers conic equal-area projection centered on the U.S., Africa, Australia, China, Europe, or South America, as well as the Mercator projection. For world maps, the Robinson and Gall-Peters projections are available. Map elements such as map title, north arrow, projection label, description, and custom place labels are also available under base map settings (see Appendices Section 3).

3.5. Computational tools

FlowMapper currently supports two functions for processing flow and location data: ‘Polygons to Points’ and ‘Normalize Flows’. ‘Polygons to Points’ tool transforms a polygon JSON file to a node (point) csv file with latitude and longitude coordinates of the geometric centroid of each polygon. ‘Normalize Flows’ tool takes an input flow csv file and transforms it into a modularity flow file using a null expectation model. Flow normalization helps remove the effect of size differences and the ability for nodes to generate flows by calculating the difference between the actual flow and the expected volume of flow for each pair of locations (nodes). We explain the process for transforming raw flows to modularity flows in the Appendices Section 2.
4. Map use cases

We demonstrate FlowMapper using three scenarios that portray distinct flow map data sets, map extents, scales, and layouts.

4.1. Scenario 1: banana trade within South America in 2019

Figure 3 illustrates banana trade flows between countries in South America from 2019 (FAO, 2021). The choropleth map illustrates the net banana trade ratio, which is calculated by:

\[
\text{Net trade ratio (i)} = 100,000 \times \frac{\text{import}(i) + \text{export}(i)}{\text{population}(i)}
\]

For each country i, we divide the sum of its imports and exports in tons by its population and multiply the division with 100,000 to derive the net banana trade ratio in tons per 100 thousand population. The countries with a negative net trade ratio are depicted in purple hue, which indicates larger export of bananas than imports in tons. On the other hand, the countries with a positive net trade ratio are illustrated in green, which imports more bananas than they export. We visualize the gross volume of the banana trade in tons (import + export) using a graduated node (point) symbol map with three classes, which are placed at approximately the centroid of country polygons. Finally, we visualize the banana trade from a country to another country using graduated flow symbols with four classes. Countries with a warmer climate that are closer to the equator export more bananas to countries in the south of the continent that have less suitable climate for producing bananas. From the nodes showing the gross volume of banana trade, we can see that Ecuador exports more bananas to other countries than Brazil and Peru while countries like
Argentina and Chile are the major importers of bananas within South America.

### 4.2. Scenario 2: family migration in the United States between 1887 and 1924

Figure 4 illustrates the number of families migrated between states from 1887 to 1924 whose data were derived from a population-scale family tree data set (Koylu et al., 2021). The choropleth map depicts migration efficiency, which represents the net migration of the state (out-migrant families subtracted from in-migrant families) divided by the sum of in and out-migrant families. A divergent color scheme is used to distinguish the net importer states (blue) from the net exporter states (orange). The graduated node symbols represent the gross volume of migrant families, which is the sum of in- and out-migrant families per state. Finally, the graduated flow symbols represent the number of families migrating between pairs of states. Predominant flows are between adjacent states and from East to West. Particularly, Oklahoma and California appear to have a great inflow of families during this period, most likely because of the oil being discovered in 1897 (Aoghs.org Editors, 2020) and the westward expansion.

### 4.3. Scenario 3: bike trips in Chicago in April 2021

Figure 5 illustrates bike trips in Chicago in April 2021 (Divvy, 2021). Node symbols are placed at bike stations and classified to represent the gross volume of flows (the sum of pick-ups and drop-offs) to distinguish how each bike station is used. The Lake Shore Dr. and Monroe St. bike station is selected to highlight flows from and to this station while the rest of the flows are faded with increased transparency. The figure uses a dark basemap with light and neon colors for nodes and flows to provide increased contrast. The selected station is nearby attractions such as Maggie Daley Park, the Art Institute of Chicago, and Cloud Gate. The major in-flows to this station come from the Chicago Field Museum, the Shedd Aquarium, and the Alder Planetarium from the southern part of downtown. The largest number of bike trips for this station are from and to the Navy Pier bikeshare station which is to the northwest. Other larger out-flows from Lake Shore Dr. and Monroe St. station are located along a bike path that goes along Lake Shore Dr. that is to the northwest of this location.

### 5. Conclusions and future work

Our overarching goal for FlowMapper is to provide users greater customizability and design flexibility that could result in an interactive map and a static map product that requires minimal touch-up in other graphics software. Our goal is to address current limitations and make enhancements so that FlowMapper can be used in the professional cartographic workflow to make polished static and interactive maps. In this article, we summarized the main contributions of FlowMapper. First, users can upload and process their own data to produce, customize and share flow maps. The ability to save data, cartographic design and map elements in a project file allows users...
to easily share their data and/or cartographic design with others. Second, users can customize the flow line symbology by including options to change the flow line style, width, and coloring. We introduce algorithms for drawing curved line styles with varying thickness along a flow line, which reduces the visual cluttering and overlapping by tapering flow lines at origin and destination points. The ability to customize flow symbology supports a range of flow map-reading tasks such as comparing flow magnitudes and directions and identifying flow and location clusters that are strongly connected with each other. Third, users can add and customize supplementary layers such as node symbol, choropleth, and base maps to support contextualizing flow patterns with location references and characteristics such as gross flow, net-flow ratio or a locational attribute such as population density and income. FlowMapper also supports user interactions to zoom, filter, and obtain details-on-demand functions to acquire information on flows, nodes, and regions. Finally, the web-based architecture of FlowMapper supports server-side computational capabilities to process and summarize large flow data and reveal natural patterns of flows.

There are several future directions we plan to pursue and address the limitations of FlowMapper. Asymmetric curvature introduced in this manuscript reduces overlapping at nodes by tapering the flow curves at origins and destination ends. We apply the same arbitrary curvature to all flows, which has been found to increase visual clutter in certain cases (Xu et al., 2012) such as drawing flows from geographically close origins to geographically close destinations. Thus, each algorithm for drawing curved flows may produce visual clutter elsewhere in a flow map layout and may violate different design principles. To give users more flexibility to choose flow line algorithms based on the desired map-reading tasks, we plan to implement more flow line designs including origin–destination coloring and partial lines, symmetric curves and curves with varying and interactively controllable symmetry, and the force-directed algorithm (Jenny et al., 2017). We also plan to implement undirected and one-way directed (or net) flow maps as

Figure 5. Selected bike trips in Chicago in April 2021 from and to Lake Shore Dr. and Monroe St. bike station using the curve with half-arrow style.
alternative map types. We plan to expand the sharing of flow maps by enabling users to host and process data and share their flow maps on the cloud. We currently support a limited number of map projections. We plan to incorporate more projections and the ability to automatically select an appropriate projection based on the spatial extent of the data. While we currently support flow normalization, we plan to integrate computational capabilities such as multi-scale flow mapping (Zhu et al., 2019), and regionalization (Guo, 2008). FlowMapper does not support the visualization of temporal flow data. We plan to extend FlowMapper to incorporate alternative time-series flow visualizations such as flow map animations and small multiples.

Software

The front end of FlowMapper is built with HTML, CSS, JavaScript and the Bootstrap template. OpenLayers is used for the mapping framework including the base maps and basic interactive map functions, and D3 (Data-Driven Documents) (Bostock et al., 2011) is used to create the interactive maps that include polygon, point and flow symbol vectors. FlowMapper utilizes a Java back-end design connected with a PostgreSQL/PostGIS database with an Apache Tomcat server. Currently, flow data transformation and normalization features can be employed using the back-end server. Back-end design will enable future data hosting, and flow data processing such as regionalization (Guo, 2008) and multi-scale flow mapping (Zhu et al., 2019).

Open scholarship

This article has earned the Center for Open Science badges for Open Data, Open Materials and Preregistered. The data and materials are openly accessible at http://doi.org/10.6084/m9.figshare.14593380.v3 and http://doi.org/10.6084/m9.figshare.14593380.v3

Acknowledgements

The authors thank Daniel Stephen, Drs. Brooke Marston, Yalong Yang, and Bernhard Jenny for their valuable feedback and suggestions. The authors also thank the Department of Geographical and Sustainability Sciences at the University of Iowa for supporting the development of FlowMapper as a teaching material to be used in geography, GIS, and other relevant courses. The authors thank Mert Erdemir, Beichen Tian and Hoeyun Kwon for their kind support and feedback.

Disclosure statement

No potential conflict of interest was reported by the authors.

Data availability statement

All the source code to produce both back-end and front-end development of FlowMapper and the example flow data sets used in this article are publicly available at: https://doi.org/10.6084/m9.figshare.14593380.v3. The data sets used in this study do not contain human subjects, and they are in the public domain. Banana trade flow data between countries in South America from 2019 were derived from Food and Agriculture Organization of United Nations website: http://www.fao.org/faostat/en/#data/TM (FAO, 2021). The country polygons were obtained from Natural Earth Data (Natural Earth, 2021). Historical family migration data between states from 1887–1924 were derived from a population-scale family tree data set produced by Koylu et al. (2021). Bike trip data in Chicago throughout the month of April 2021 were derived from https://www.divvybikes.com/system-data (Divvy, 2021). All the source code and materials for FlowMapper are also published on this GitHub link: https://github.com geo-social/flowmapper

ORCID

Caglar Koylu © http://orcid.org/0000-0001-6619-6366

References

Andrienko, G., Andrienko, N., Fuchs, G., & Wood, J. (2017). Revealing patterns and trends of mass mobility through spatial and temporal abstraction of origin-destination movement data. IEEE Transactions on Visualization and Computer Graphics, 23(9), 2120–2136. https://doi.org/10.1109/TVCG.2016.2616404

Aoghs.org Editors. (2020). Oklahoma oil history. https://aoghs.org/oil-almanac/oklahoma-oil-history/

Bostock, M., Ogievetsky, V., & Heer, J. (2011). D³ data-driven documents. IEEE Transactions on Visualization and Computer Graphics, 17(12), 2301–2309. https://doi.org/10.1109/TVCG.2011.185

Boyandin, I. (2021). flowmap.blue. https://flowmap.blue

Boyandin, I., Bertini, E., Bak, P., & Lalanne, D. (2011). Flowstrates: An approach for visual exploration of temporal origin-destination data. Paper presented at the Computer Graphics Forum.

Dent, B., Torguson, J., & Hodler, T. (2008). Thematic map design. McGraw-Hill.

Divvy. (2021). Historical bike trip data. https://www.divvybikes.com/system-data

Dong, W., Wang, S., Chen, Y., & Meng, L. (2018). Using eye tracking to evaluate the usability of flow maps. ISPRS International Journal of Geo-Information, 7(7), Article 281. https://doi.org/10.3390/ijgi7070281

FAO. (2021). Food and Agriculture Organization of United Nations (FAO). http://www.fao.org/faostat/en/#data/TM

Ghoniem, M., Fekete, J.-D., & Castagliola, P. (2005). On the information systems. Information Visualization, 4(2), 114–135. https://doi.org/10.1057/palgrave.ivs.9500692

Graser, A., Schmidt, J., Roth, F., & Brändle, N. (2019). Untangling origin-destination flows in geographic information systems. Information Visualization, 18(1), 153–172. https://doi.org/10.1177/1473871617738122

Guo, D. (2008). Regionalization with dynamically constrained agglomerative clustering and partitioning (REDCAP).
International Journal of Geographical Information Science, 22 (7), 801–823. https://doi.org/10.1080/13658810.701764970

Guo, D. (2009). Flow mapping and multivariate visualization of large spatial interaction data. IEEE Transactions on Visualization and Computer Graphics, 15(6), 1041–1048. https://doi.org/10.1109/TVCG.2009.143

Guo, D., & Zhu, X. (2014). Origin-destination flow data smoothing and mapping. IEEE Transactions on Visualization and Computer Graphics, 20(12), 2043–2052. https://doi.org/10.1109/TVCG.2014.2346271

He, S. (2018). From beautiful maps to actionable insights: Introducing Kepler.gl, Uber’s open source geospatial toolbox. Uber Engineering, 29.

Holten, D., Isenberg, P., Van Wijk, J. J., & Fekete, J. D. (2011). Force-directed edge bundling for graph visualization. Computer Graphics Forum, 28(3), 983–990. https://doi.org/10.1111/j.1467-8659.2009.01450.x

Jenny, B., Stephen, D. M., Muehlenhaus, I., Marston, B. E., Sharma, R., Zhang, E., & Jenny, H. (2017). Force-directed layout of origin-destination flow maps. International Journal of Geographical Information Science, 31(8), 1521–1540. https://doi.org/10.1080/13658816.2017.1307378

Jenny, B., Stephen, D. M., Muehlenhaus, I., Marston, B. E., Sharma, R., Zhang, E., & Jenny, H. (2018). Design principles for origin-destination flow maps. Cartography and Geographic Information Science, 45(1), 62–75. https://doi.org/10.1080/15230406.2016.1262280

Koylu, C. (2014). Understanding geo-social network patterns: Computation, visualization, and usability [Ph.D.]. University of South Carolina. ProQuest Dissertations Publishing.

Koylu, C., & Guo, D. (2017). Design and evaluation of line symbolizations for origin-destination flow maps. Information Visualization, 16(4), 309–331. https://doi.org/10.1177/1473871616681375

Koylu, C., Guo, D., Huang, Y., Kasakoff, A., & Grieve, J. (2021). Connecting family trees to construct a population-scale and longitudinal geo-social network for the U.S. International Journal of Geographical Information Science, 35(12), 2380–2423. https://doi.org/10.1080/13658816.2020.1821885

Natural Earth. (2021). Country shapefiles. https://www.naturealearthdata.com

Nost, E., Rosenfeld, H., Vincent, K., Moore, S. A., & Roth, R. E. (2017). Hazmatmapper: An online and interactive geographic visualization tool for exploring transnational flows of hazardous waste and environmental justice. Journal of Maps, 13(1), 14–23. https://doi.org/10.1080/17445647.2017.1282384

Parisi, T. (2012). WebGL: Up and running. O’Reilly Media, Inc.

Phan, D., Xiao, L., Yeh, R., & Hanrahan, P. (2005). Flow map layout. Paper presented at the IEEE Symposium on Information Visualization.

Slocum, T. A., McMaster, R. B., Kessler, F. C., & Howard, H. H. (2009). Thematic cartography and geovisualization. Pearson Prentice Hall.

Stephen, D. M., & Jenny, B. (2017). Automated layout of origin-destination flow maps: US county-to-county migration 2009–2013. Journal of Maps, 13(1), 46–55. https://doi.org/10.1080/17445647.2017.1313788

Tao, R., & Thill, J. C. (2016). Spatial cluster detection in spatial flow data. Geographical Analysis, 48(4), 355–372. https://doi.org/10.1111/gean.12100

Tobler, W. R. (1987). Experiments in migration mapping by computer. American Cartographer, 14(2), 155–163. https://doi.org/10.1559/15230408778375273

Vincent, K., Roth, R. E., Moore, S. A., Huang, Q., Lally, N., Sack, C. M., Nost, E., & Rosenfeld, H. (2019). Improving spatial decision making using interactive maps: An empirical study on interface complexity and decision complexity in the North American hazardous waste trade. Environment and Planning B: Urban Analytics and City Science, 46(9), 1706–1723. https://doi.org/10.1177/239980318764122

von Landesberger, T., Brodkorb, F., Roskosch, P., Andrienko, N., Andrienko, G., & Kerren, A. (2016). Mobilitygraphs: Visual analysis of mass mobility dynamics via spatio-temporal graphs and clustering. IEEE Transactions on Visualization and Computer Graphics, 22(1), 11–20. https://doi.org/10.1109/TVCG.2015.2468111

Ware, C., Kelley, J. G., & Pilar, D. (2014). Improving the display of wind patterns and ocean currents. Bulletin of the American Meteorological Society, 95(10), 1573–1581. https://doi.org/10.1175/BAMS-D-13-00135.1

Wood, J., Dykes, J., & Slingsby, A. (2010). Visualisation of origins, destinations and flows with OD maps. Cartographic Journal, 47(2), 117–129. https://doi.org/10.1177/0008704101258023467367

Xu, K., Rooney, C., Passmore, P., Ham, D.-H., & Nguyen, P. H. (2012). A user study on curved edges in graph visualization. IEEE Transactions on Visualization and Computer Graphics, 18(12), 2449–2456. https://doi.org/10.1109/TVCG.2012.189

Yang, Y., Dwyer, T., Goodwin, S., & Marriott, K. (2017). Many-to-many geographically-embedded flow visualisation: An evaluation. Ieee Transactions on Visualization and Computer Graphics, 23(1), 411–420. https://doi.org/10.1109/TVCG.2016.2598885

Zhu, X., Guo, D., Koylu, C., & Chen, C. (2019). Density-based multi-scale flow mapping and generalization. Computers, Environment and Urban Systems, 77, Article 101359. https://doi.org/10.1016/j.compenvurbsys.2019.101359