Arc and Swarm-based Representations of Customer's Flows Among Supermarkets

Evgheni Polisciuc[1], Pedro Cruz[1], Hugo Amaro[1], Catarina Mas[1], Tiago Carvalho[2], Frederico Santos[2], Penousal Machado[1]

[1]*CISUC, Department of Informatics Engineering, University of Coimbra, Portugal*

[2]*Sonae, Portugal*

cmacas@student.dei.uc.pt, pmcruz@dei.uc.pt, hamaro@student.dei.uc.pt, evgheni@dei.uc.pt, {frederico.santos, tiago.carvalho}@sonae.pt, machado@dei.uc.pt

Keywords: OD visualization, flow visualization, geo-temporal visualization, big data

Abstract: Representing large amounts of flows involves dealing with the representation of directionality and the reduction of visual cluttering. This article describes the application of two flow representation techniques to the visualization of transitions of customers among supermarkets over time. The first approach relies in arc representations together with a combination of methods to represent directionality of transitions. The other approach uses a swarm-based system in order to reduce visual clutter, bundling edges in an organic fashion and improving clarity.

1 INTRODUCTION

Nowadays data are collected faster than analyzed. The advances in computation allow the storage and processing of large amounts of data in a reasonable time. Additionally, current visualization techniques enable efficient analysis of data, while trying to deal with generated visual clutter in order to achieve better visual clarity. Graphical exploratory analysis of transitions in space is important to many fields of study, since it enables the understanding of the amounts of flows among geographical locations, while enabling to focus on certain geographical areas and time spans.

In this article we present an application of visual techniques to represent big amounts of consumption data in more than 700 supermarkets and hypermarkets in Portugal, having all registered transactions from May of 2012 to April of 2014. This article focuses on the graphical exploration of customers transitions among supermarkets over time. Transitions are visualized in order to reveal flow patterns of customers. We considered a transition whenever there is a change in a transaction's location in a customer's shopping history. These transitions can be represented as origin-destination (OD) vectors. The reasons that cause a transaction can be fairly intricate, but can perhaps be related with local and temporal discounts, and seasonal and definite changes in a customer's residency.

This paper tackles the visualization of flows of transitions using two approaches. More precisely, the issue of visual clutter in high-dense representation and the directionality of flow streams. The first approach employs a well known visualization technique, which is based on arc representation (section 3 provides a detailed description of the application of this technique to our data). The second approach relies on bio-inspired mechanism known as flocking. The application of this technique enables the representation to convey information on transition flows with bio-inspired aesthetics while reducing the amount of visual clutter to improve clarity (see section 4 for a detailed description).

2 BACKGROUND AND RELATED WORK

Arc diagrams are a widely known method to visualize structures in text, songs or any other sequences of symbols popularized by Wattenberg M. (Wattenberg, 2002). Since then, arcs have been applied in different domains, namely in graph visualization and geographic visualization, particularly to represent origin-destination data. The work of Schich et al. (Schich et al., 2014) is an example of application of arcs in geographic context to represent OD-like data. In this work origin and destination represents, respectively, the place of birth and place of death of notable people in the history. The directionality is represented with color interpolation (red-blue for origin-destination). One of the concerns of origin-destination visualization is the representation of directionality of edges, particularly when dealing with bidirectional flows. Recent work (Holten and van Wijk, 2009) has presented six different ways of edge directionality representation (tapered, dark-to-light, light-to-dark, arrow, curved, and green-to-red) and compared the reading performance of each technique. This study suggests that the tapered method is advantageous in most situations, unlike curved representation which is the worst of all cases. In any cases, the representation of bidirectional data is still challenging, due to additional visual information added to each edge.

Direct visualization of large volumes of OD data generates high degrees of visual clutter. In these cases a reduction strategy known as edge bundling can be applied, which is characterized not only by graph simplification, but also by the revelation of principal streams of flow. Holten introduced edge bundling for compound graphs. His work consisted of routing edges through a hierarchical layout using B-Splines (Holten, 2006). There are several variations of edge bundling starting with force-directed (Holten and Van Wijk, 2009) up to sophisticated kernel density estimation strategies (Hurter et al., 2012). Generally, edge bundling consists of drawing similar edges on the same path, i.e. edges that are related in geometry and direction are routed along the same path.

In the geographic context OD representation as a rule refers to the flow visualization (also known as flow maps), which is deeply rooted in the history of information visualization. Early examples, such as wine exports from France, produced by Minard (Tufte and Graves-Morris, 1983, page: 25), represents quantity as well as direction of wine exports encoded by the thickness of the corresponding edges, which disjoin from the parent edge. The work of Phan et al. (Phan et al., 2005) describes an automated approach to the generation of flow maps using a hierarchical clustering algorithm, given a series of nodes and flow data. Generally, in geographic context flow visualization refers to the representation of amounts of any type of variables that move from one location to another (e.g. migrations, transportation of goods, etc.). The advantage of flow maps is that they reduce visual clutter by merging edges. However, they present a series of of problems, such as the perception of directionality of flow, when large amounts of bidirectional OD data is considered.

3 DATA DESCRIPTION

Our dataset consists of 278GB of information about customer purchases in 729 supermarkets and hypermarkets in Portugal in a time span of 24 months (from May, 2012 until April, 2014), including the geo localization of 682 supermarkets, as well as the regions of the country they belong to. The dataset comprises approximately 2.86 billions of transactions where each transaction has the following attributes: customer card id, amount spent, product designation, quantity of the purchased products and the date and time of the transaction. It is important to note that several individuals may hold the same customer card with an unique client id (e.g. members of a family). The dataset has a total of 6.6 Million unique card ids.

Before the extraction of transitions among supermarkets we first compute their geographical clusters. The reason for that is because the majority of supermarkets belong to shopping centers which are considered as a unique geographical location. In this case the DBSCAN algorithm (Ester et al., 1996) was applied with the parameters of 0 for K and 0.01 for epsilon. As a result 304 clusters were obtained, where the extracted locations are the centroids of the clusters of supermarkets (each centroid will be referenced as a single supermarket for the sake of simplicity).

With the clusters computed we proceed to extract transitions as follows: first the data is aggregated by day (24 hours); then for each client the sequence of transitions is computed by excluding subsequences of repeated places. For example, let $X = (A, A, B, B, B, C)$ be the sequence of supermarkets where a client made transactions. So, the transition sequence would be $X_{tr} = (t_1(A, B), t_2(B, C))$.

4 ARC REPRESENTATION

Our first approach was based on direct representation of the data. The transition sequence is directly encoded by edges, that represents the link between the origin-destination supermarket, as well as the number of clients that transitioned. The directionality of the edge is represented based on the combination of taped and curved methods, due to the bidirectionally of data. Since arc-based approach usually do not represent directionality, the thickness of arcs in our visualization increase as they approach their destination, resembling the trajectory of a projectile or a comet. The asymmetrical curve gives a more natural sense of direction. Arcs where also used because they reduce visual clutter when compared with straight lines methods.

4.1 Arc Anatomy

The arcs consist of a bezier curve (Farin et al., 2002, page: 4-6) with two control points (see Figure 1). These points are the vertices of triangles $ODC1$ and $DOC2$. The two triangles are computed differently with empirically determined values. The length of the $DC1$ edge has 60% of the length of OD, and the angle β is equal to 27. The $OC2$ edge has 90% of the length of OD, and the angle α varies proportionally to the OD length, and is constrained to the range [10, 20]. The upper and lower angular limits correspond to the maximum and minimum length of the set of ODs. By adjusting the $C2$ control point the arc changes its curvature, making the long arcs visually distinct from short ones.

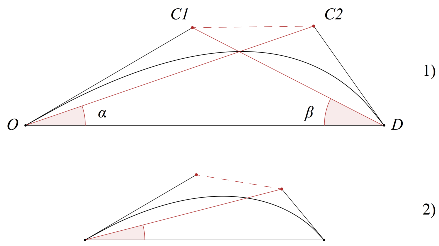

Figure 1: The position of control points for different OD distances.

The arc on its own does not convey any information besides the connection of two points. In order to encode the quantity of clients involved in the same transitions we use the thickness and transparency of the line (Figure 2). The thickness give a good estimation of the encoded value, while opacity diminishes the impact of less relevant transitions. Only the destination side of the arc changes its thickness, interpolating from the destination value (amount of transitioned clients) to a certain minimum at the opposite side of the arc. Moreover, this kind of representation gives a clear understanding of the direction of OD data (Holten and van Wijk, 2009).

The color of each arc varies with respect to the corresponding value. The minimum and maximum numbers of transitions are represented with blue and orange respectively, and intermediate colors are interpolated according to the value (Figure 2). Therefore, arcs that represent few transitions appear in transparent blue color, unlike arcs that represent high number of transitions, which appear in saturated orange color.

4.2 Application

Each day in the dataset is visualized separately displaying only the transitions that occurred on that particular day. By navigating on the timeline the user can change the current day. Although, we can zoom to any part of the country to analyze it in detail, we focused on two major metropolitan areas - Porto and Lisbon - and in a general view of the country (Figure 3). In the general view all the arcs whose origin and destination locations are inside the same region are not represented, since the density of such transitions at this zoom level does enable a proper representation beyond visual noise. Therefore, in the general view we opted to visualize only inter-regional transitions. In closer views we only display the arcs that fit in the viewing area.

Figure 2: Arcs representing the data variable. Reading top down each arc represents maximum, 75% quartile, median and 25% quartile value. The colors are interpolated in the range $RGBA[(86,150,255,30),(255,152,74,100)]$

While applying this technique to the dataset we noted several limitations. First, a large data density generates a high degree of visual clutter, making general view hard to analyze (Figure 3, image on the left). For instance edges that connect supermarkets in Lisbon and supermarkets in Porto hide a significant part of the edges that connect other urban areas, for example Coimbra and Porto. Also, in our data the most of the transitions occur within shorter distances limiting this approach in terms of analysis, since it is extremely difficult to estimate the values and direction of short arcs (See for example Figure 3, images in the middle and on the right).

5 SECOND APPROACH

In the second approach we relied on nature-inspired mechanisms, more precisely on a swarm system. Each transition in the sequence is encoded through a simulated path of a boid. By running the system each boid interacts with the ghosts of other boids (Reynolds, 1987), updating its own path at each sim-

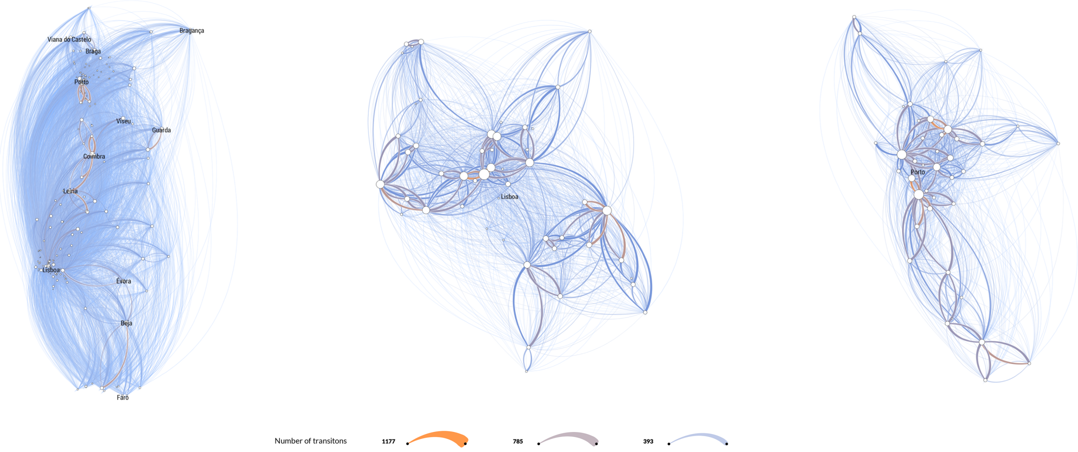

Figure 3: The general view of Portugal (left), metropolitan area of Porto (middle), metropolitan area of Lisbon (right). The displayed date is 23 of December, 2012. The scale of the zoomed views with respect to the general view is 1:10.

ulation step. Ghosts of each boid hold information about the position, direction, destination and data value at each simulation point on the path. The process is iterative and in each execution cycle all the active boids are simulated.

5.1 Flocking and Flow Representation

In order to reflect the flowing nature of the information we resort to a swarm system, which is comprised by artificial agents (boids) that react to the presence and characteristics of neighboring boids. While running the system each boid simulates the flow of data, adapting the paths that represent OD edges, bundling them and making visual patterns emerge. As a result, the visualization represents the flows in the dataset with reduced degree of visual clutter.

Each boid in the system is characterized by direction, speed, radius of vision, the number of transitions that it represents, a set of behavioral rules, and its unique origin-destination points. During the simulation each boid leaves persistent traces, further referenced as ghosts, that contain information for the speed and position at that point as well as a reference to the boid itself.

The behavioral rules of each boid are determined through the interaction with the traces of other members of the flock. Pairwise comparison between boids and ghosts establishes the relationship between them and their behavior. If the agents advance in similar directions, they are considered friendly. If the agents advance in opposite directions, they are considered unfriendly. Otherwise, they ignore each other. The degree of similarity affects the force of attraction or repulsion between agents and ghosts. Therefore, friendly agents advance together as a group and unfriendly agents repel from each other avoiding collisions. Figure 4 illustrates this behavior.

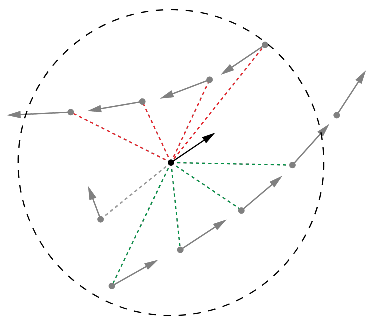

Figure 4: Pairwise comparison between one boid and neighboring ghosts. The black dot and the arrow in the center are the current boid and its direction. The gray dots are ghosts left by other boids. The dashed circle is the radius of vision and the dashed lines are the relations with the ghosts green and red lines represent friendly and unfriendly relationships, respectively. Gray dashed line connects the ghost that is ignored.

The direction and the speed of each boid B with position $\vec{p}_B$ depends on the position $\vec{p}_X$ of ghosts X within the radius of vision d_{VR} and the angle between their direction vectors $\vec{d}_X$ and $\vec{d}_B$. Each boid in the system simulates its path until reaching its destination. Otherwise, the boid finishes its simulation and is marked as inactive. The following rules are applied to each active boid in the system.

Stick with friends. Each boid attempts to move towards the center of the group of friendly ghosts. Friendly ghosts are determined by the angle between directions of the boid and neighbor ghosts. Their similarity creates stronger relationship and are determined by the distance and the angle between their directions.

$$\left.\begin{array}{c}||\vec{p_X}-\vec{p_B}||\leq d_{VR}\\ \widehat{\vec{d_X}\vec{d_B}}<a_{max}\end{array}\right\}\Rightarrow \vec{v_F}=\frac{1}{n_X}\sum_X \frac{\vec{p_X}-\vec{p_B}}{||\vec{p_X}-\vec{p_B}||}w(X,B) \quad (1)$$

Avoid unfriendly boids. Each boid attempts to avoid collision with the ghosts of other boids if the angle between their directions are bigger than $\pi - a_{max}$.

$$\left.\begin{array}{c}||\vec{p_X}-\vec{p_B}||\leq d_{VR}\\ \widehat{\vec{d_X}\vec{d_B}}<\pi-a_{max}\end{array}\right\}\Rightarrow \vec{v_F}=\frac{1}{n}\sum_X \begin{cases}-\perp \hat{d_B}w(X,B) & ifC_Z(\vec{p_X})\geq 0\\ \perp \hat{d_B}w(X,B) & ifC_Z(\vec{p_X})<0\end{cases} \quad (2)$$

Similarity between two boids determines the weight of the force, and is related to the distance between boids, the angle between their directions, and the data value, this is, the quantity of transactions that are encoded. Boids that contain higher quantities have greater impact over the other members of the flock. So, the similarity between two boids is computed as follows:

$$w(X,B)=\frac{w_a(\vec{d_X},\vec{d_B})+w_m(\vec{p_X},\vec{p_B})}{2}\times data_X \quad (3)$$

$$w_a(\vec{d_X},\vec{d_B})=1-\left(\frac{\widehat{\vec{d_X}\vec{d_B}}}{a_{max}}\right)^3 \quad (4)$$

$$w_m(\vec{p_X},\vec{p_B})=2^{(2-0.1||\vec{p_X}-\vec{p_B}||)} \quad (5)$$

Where d_X and d_B are the vectors of the direction of boid B and ghost X, and a_{max} is the maximum angle allowed between two direction vectors. dataX is the normalized value of data variable $((value - min)/(max-min))$. All the parameters were empirically determined.

Avoid static points. Every boid B in the system attempts to avoid collision with the static points S with the position $\vec{c_S}$, which is the centroid of a cluster of supermarkets. The origin and destination points do not enter in the calculation.

$$||\vec{c_S}-\vec{p_B}||\leq d_{SP}\Rightarrow \vec{v_{SP}}=\frac{1}{n}\sum_S\begin{cases}-\perp \hat{d_B} & ifC_Z(\vec{C_S})\geq 0\\ \perp \hat{d_B} & ifC_Z(\vec{C_S})<0\end{cases} \quad (6)$$

$$C_Z(\vec{p})=\left(\frac{\vec{p}-\vec{p_B}}{||\vec{p}-\vec{p_B}||}\times \vec{d_B}\right)_Z \quad (7)$$

Finally, each boid is always attracted by its destination, having the force vector equal to the normalized vector pointing towards the boids destination. When the boid approximates its destination, all the forces, except the destination force, are ignored and the speed is limited to 1. This restriction ensures that each boid reaches its destination.

The location of boids is defined by applying the computed vector forces as follows: all the forces are added to the acceleration; then the acceleration is added to the speed; finally, the speed is limited to the predefined maximum and is added to the current location. The maximum defined speed reflects on the visual output resulting in high and low curvature of edges for speed limited to 1 and 3, respectively.

In each iteration the boid's paths are updated according to the current state of the system. More precisely, during the execution cycle each boid interacts with the ghosts left by other boids and never with their own ghosts. The process stops when the boid reaches its destination or its path remain unaltered during last three simulation steps. In order to determine if a path has changed since the last iteration we compute the root-mean-square deviation (RMSD) at each iteration. So, given the current path P_C and the previous path P_P which comprise sequences of ghosts P computed in current and the last iteration, $RMSD$ is calculated as follows:

$$RMSD=\sqrt{\frac{1}{n}\sum_n^i ||P_{Ci}-P_{Pi}||^2} \quad (8)$$

Where n is the minimum number of ghosts in both paths. If the average of the last three $RMSD$ is below a certain threshold (in our case 0.5) then the boid is marked as inactive preventing it from further updating of its path.

The application of this technique to the data has similar aspects with the first approach - focus on the general view and two zoomed views for Lisbon and Porto metropolitan areas. In the general view all the supermarkets that belong to the same region are not considered, while in the zoomed views the edges that do not fit in the zoomed area are not computed. The number of transitions per path is encoded by the thickness of the line and by the color scheme used in arc representation. The directionality is represented using the tapered method (see Fig. 5).

At this stage we found limitations in our approach. First there is no guaranty that the algorithm converges, since it highly depends of the overall state of the system, which varies according to the data. For that reason we established a threshold equal to 99.9% of inactive boids to force the algorithm to stop. The complexity of the algorithm also depends on data, more precisely it depends on the number of OD edges and their length, since each boid computes its position by considering the trails of every other boid and more lengthy edges implies that more trails are generated.

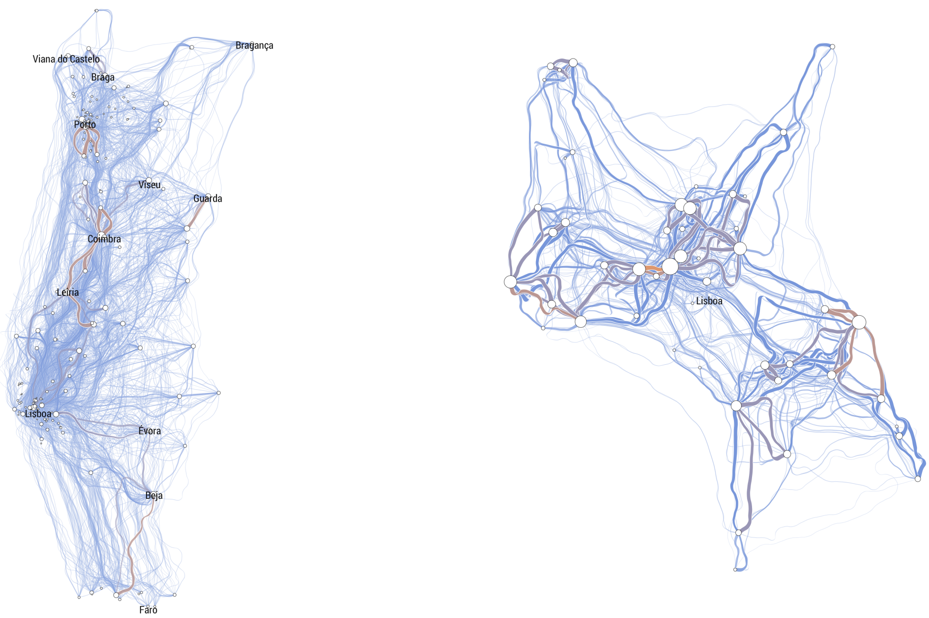

Figure 5: General view of Portugal, image on the left, and metropolitan area of Lisbon, image on the right. The displayed date is 23 of December, 2012.

6 DISCUSSION

In this section, arc representation, swarm-based representation and force directed edge bundling (FDEB) are compared and discussed. The visual output from the three techniques is displayed in Figure 6. To compare the approaches we choose a day before Christmas (23 of December of 2012) and zoomed on Lisbon's view. The same visual mapping is applied in all three approaches allowing a fair comparison between them. The usage of the color and the thickness of edges was described in previous sections.

The very first comparison reveals the efficiency of visual clutter reduction. As can be observed, the force directed edge bundling method generates less visual clutter in comparison with arc and swarm-based representation. Also, swarm-based visualization is less cluttered than arc representation. When using swarms, main streams of flow are visually distinct from each other leaving enough space for the ones with less impact.

In the swarm system each boid attempts to avoid the boids with opposite directions, as such the simulated paths are never routed through the same trail, making it possible to distinguish paths that encode opposite directions. As can be observed, this isn't the case when using the FDEB, and the arc methods. These algorithms do not take into account the directionality of streams, which is an emergent characteristic of the swarm-based approach. Finally, since the boids in the system attempt to avoid static points, the Supermarkets, nodes encoded with white circles, are clearly visible and do not visually interfere with the lines drawn by the swarming algorithm.

7 CONCLUSIONS AND FUTURE WORK

In this article we have presented two graphical explorations of transitions among supermarkets. In the first approach we explored direct representation of transition sequences. This consists of the combination of curved and taped strategies to represent origin-destination data, which enables the perception of bidirectional edges. The shape of the arcs, inspired on the trajectory of launched projectiles, enforces the readability of direction. The number of clients, whose transition is represented by an arc, is encoded by the line thickness, and by color.

The second approach overcomes the cluttering issue in the visualization compared to the first one. This approach consists of a set of boids that represent each transition with their paths. Each singe boid follows simple behavioral rules by pairwise interaction with other neighbor boids. When two neighboring boids advance in the similar direction they are considered as friends and attempt to move together. In contrast, when two neighboring boids have opposite directions

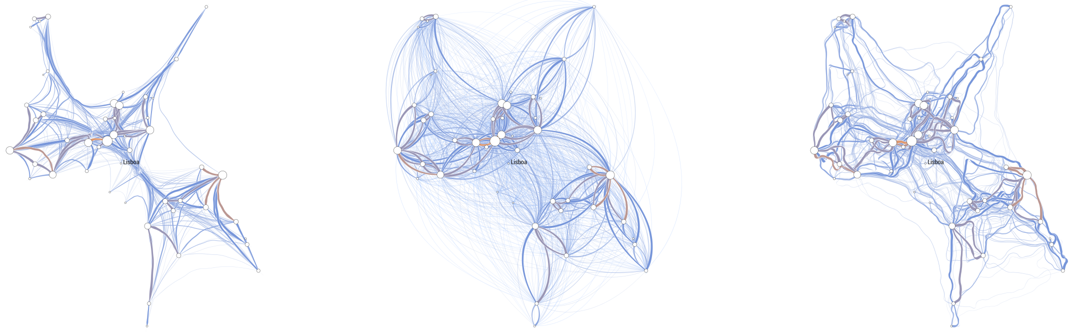

Figure 6: Three approaches for representing OD data. Force directed edge bundling (left), arc-based representation (middle) and swarm-based representation (right). Displayed metropolitan area of Lisbon on 23 of December of 2012.

they are considered as not friends and attempt to avoid each other. Otherwise, they ignore each other. The relation between two boids is determined by the distance between them and the angle between their directions. The boids that represent more customers have higher impact on other members of the system. Finally, every boid attempts to avoid static points, except when these are located nearby the origin or the destination point.

The arcs visualization for large volumes of origin-destination data generates high degree of visual clutter. In contrast, our swarm-based approach simplified the visualization representing the flow data in a natural and organic manner, but is computationally intensive when high volumes of data are considered. The force directed edge bungling method generates even less visual clutter in comparison with our approaches. However, the swarm-based representation visually separates the streams of flow with opposite directionality, which does not happens in force directed edge bundling.

As future work we will improve the performance of our swarm-based approach, for example by using a quadtree structure to store and gain faster access to nearby ghost boids. In order to further improve the efficiency of the technique, the algorithm can update existing ghost boids with new information instead of indefinitely adding new ghost boids to the same quadtree cell.

ACKNOWLEDGEMENTS

This research is partially funded by: iCIS project (CENTRO-07-ST24-FEDER-002003), which is co-financed by QREN, in the scope of the Mais Centro Program and European Union's FEDER; Sonae Viz Big Data Visualization for retail.

REFERENCES

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231.

Farin, G. E., Hoschek, J., and Kim, M.-S. (2002). *Handbook of computer aided geometric design*. Elsevier.

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

Holten, D. and Van Wijk, J. J. (2009). Force-directed edge bundling for graph visualization. In *Computer Graphics Forum*, volume 28, pages 983–990. Wiley Online Library.

Holten, D. and van Wijk, J. J. (2009). A user study on visualizing directed edges in graphs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2299–2308. ACM.

Hurter, C., Ersoy, O., and Telea, A. (2012). Graph bundling by kernel density estimation. In *Computer Graphics Forum*, volume 31, pages 865–874. Wiley Online Library.

Phan, D., Xiao, L., Yeh, R., and Hanrahan, P. (2005). Flow map layout. In *Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on*, pages 219–224. IEEE.

Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 21(4):25–34.

Schich, M., Song, C., Ahn, Y.-Y., Mirsky, A., Martino, M., Barabási, A.-L., and Helbing, D. (2014). A network framework of cultural history. *science*, 345(6196):558–562.

Tufte, E. R. and Graves-Morris, P. (1983). *The visual display of quantitative information*, volume 2. Graphics press Cheshire, CT.

Wattenberg, M. (2002). Arc diagrams: Visualizing structure in strings. In *Information Visualization, 2002. INFOVIS 2002. IEEE Symposium on*, pages 110–116. IEEE.