J Vis
https://doi.org/10.1007/s12650-022-00878-z

REGULAR PAPER

Zhiwei Wei · Wenjia Xu · Su Ding · Song Zhang · Yang Wang

Efficient and stable circular cartograms for time-varying data by using improved elastic beam algorithm and hierarchical optimization

Received: 1 August 2022 / Accepted: 9 September 2022

Abstract The circular cartogram, also known as the famous DorlingMap, is widely used to visualize geographical statistics by representing geographical regions as circles. However, all existing approaches for circular cartograms are only designed for static data. While applying these approaches for time-varying data, the circle locations in each circular cartogram are recomputed separately and will result in low efficiency and low visual stability between sequential circle cartograms. To generate visually stable circular cartograms for time-varying data efficiently, we propose a novel approach by improving the elastic beam algorithm with a hierarchical optimization strategy. First, the time-varying data at different time points are grouped using a hierarchical clustering method based on their similarity, and a hierarchy is then built for their corresponding circular cartograms. Second, we generate intermediate circle locations level by level for clusters of circular cartograms according to the built hierarchy with an improved elastic beam algorithm iteratively. The elastic beam algorithm is improved in its proximity graph construction and force computation by considering that the algorithm will be applied to displace circles in a cluster of circular cartograms. The iterative process stops until we obtain satisfactory circular cartograms for each time point. The evaluation results indicate that the proposed approach can achieve a higher quality (184.85%↑ and 265.69%↑) on visual stability, and a higher efficiency (58.54%↑ and 73.96%↑) with almost the same quality on overlap avoidance and relation maintenance by comparing to the existing approaches. Project website: https://github.com/TrentonWei/DorlingMap.

Z. Wei (✉) · Y. Wang
School of Remote Sensing and Geomatics Engineering, Nanjing University of Information Science and Technology, Nanjing, China
E-mail: 2011301130108@whu.edu.cn

Y. Wang
E-mail: Primular@163.com

Z. Wei · Y. Wang
Aerospace Information Research Institute, Key Laboratory of Network Information System Technology (NIST), Institute of Electronics, Chinese Academy of Sciences, Beijing, China

W. Xu
Beijing University of Posts and Telecommunications, Beijing, China
E-mail: xuwen-jia16@mails.ucas.ac.cn

S. Ding
College of Environmental and Resource Science, Zhejiang A&F University, Hangzhou, China
E-mail: suding@zafu.edu.cn

S. Zhang
School of Electronic, Electrical and Communication Engineering, University of the Chinese Academy of Sciences, Beijing, China
E-mail: zhangsong20@mails.ucas.ac.cn

Published online: 26 September 2022

Keywords DorlingMap · Time-varying data · Hierarchical clustering · Population mapping · Energy minimization

1 Introduction

The area cartogram is a powerful visualization tool for geographical statistics (e.g., population, election, etc.), in which the geographical regions are reshaped with their size in proportion to the statistical values (Nusrat and Kobourov 2016). According to their used shapes, the area cartograms can be classified into complex shape based (Chrisman 1997; Dougenik et al. 1985; Jackel 1997; Gastner and Newman 2004; Wolf 2005; Sun 2013a, 2013b, 2020; Gastner et al. 2018; Kronenfeld 2018) and regular shape based (Inoue 2011; Raisz 1934; Dorling 1996; Heilmann et al. 2004; Speckmann et al. 2006; Van Kreveld and Speckmann 2007; Buchin et al. 2012; Tang 2013; Wei et al. 2022). The circle, as one of the simplest shapes, is a widely used shape in regular shape-based area cartograms. By comparing to other kinds of area cartograms, the area cartograms that use circles are easier for users to perceive the quantity difference and are known as the circular cartogram or the famous DorlingMap (Inoue 2011; Dorling 1996).

To produce a circular cartogram, the circles representing the geographical regions are first generated in the centers of their represented regions with their sizes in proportion to the statistical values. Because the regions are converted into circles, distortions in the relations between the regions and overlaps between these circles may occur. These initially generated circles need to be displaced to new locations with quality constraints such as relations between regions maintenance and overlap avoidance. Because the initial circles can be easily generated according to the statistical values. The circle displacement is the key process for automated circular cartogram production, and many approaches are developed. By considering all involved quality constraints in a local view, the approaches which simulate the orbits of stars and planets to displace circles step by step are proposed (Dorling 1996; Tang 2013). But these local view-based approaches will result in local optimization and relations between regions may not be well maintained. Then some approaches to achieve global optimization of involved quality constraints are proposed, such as the least-square adjustment and the elastic beam algorithm (Inoue 2011; Wei et al. 2022). Due to the popularity of circular cartograms, some JavaScript libraries for data visualization such as D3 and Protovis also integrate these approaches (Bostock et al. 2011; Protovis – Dorling Cartograms 2010).

However, the above-mentioned approaches are all designed for static data. With the rapid development of information technology, spatial–temporal statistics nowadays are mostly available, e.g., the population with a time series (Castells and Blackwell 1998). The multi-view circular cartograms by using a timeline or a calendar form to display the time-varying data are also widely used (Ying et al. 2021). If the above-mentioned approaches are applied for time-varying data visualization, the multi-views (circular cartograms for each time point) need to be recomputed separately which is not efficient enough. Furthermore, visual stability is an important quality constraint for time-varying data visualization to keep the visual consistency of corresponding objects at sequential views (Chi et al. 2015; Sondag et al. 2017). But the relations between sequential circular cartograms are not considered if each circular cartogram is recomputed separately, and the visual stability between sequential views is hard to be guaranteed.

Naturally, similar data at different time points will have similar circular cartograms. Thus, the circle locations in these similar circular cartograms are also close to each other. Then the resultant locations of circles in similar circular cartograms can be obtained by displacing them from closer intermediate locations, but not from their initial locations(center of their represented regions). To this end, we can generate closer intermediate locations for similar circular cartograms first. The resultant circle locations in similar circular cartograms can then be obtained by displacing them from the intermediate locations with smaller displacement distances. This can help maintain the visual stability between circular cartograms. Furthermore, most existing approaches for circle displacement are implemented by using an iterative process (Dorling 1996; Tang 2013; Wei et al. 2022; Bostock et al. 2011; Protovis – Dorling Cartograms 2010), and smaller displacement distance also means fewer iterative steps and will improve the efficiency.

Motivated by the above thoughts, we try to produce visually stable circular cartograms efficiently for time-varying data by generating intermediate circle locations according to the similarity between data. We group similar data at different time points using a hierarchical clustering method, and a hierarchy is built for their corresponding circular cartograms. Then we generate different levels of intermediate circle locations for the clusters of similar circular cartograms according to the built hierarchy with an improved elastic beam algorithm iteratively. The iterative process stops until satisfactory circular cartograms for each time point

are obtained. Because the current elastic beam algorithm is designed for static data, we also improve the algorithm in its proximity graph construction and force computation by considering the potential displacement of circles in a cluster of similar circular cartograms.

2 Related works

It has been a long history to visualize geographical statistics by using circular cartograms. One of the earliest circular cartograms can be traced back to 1897 in *The Rand McNally World Atlas* (Reyes Nunez 2014). Due to their clear insight into the spatial distribution of geographical statistics, the circular cartograms were quickly applied in magazines and newspapers to visualize population, energy consumption, gross domestic product, etc.

The early circular cartograms were mostly produced by hand, and automatic approaches were introduced with the developments of computer science. The automated circular cartogram production is usually divided into two steps: (1) Initial circles generation: the circles are first generated in the centers of their represented regions according to the statistical values; (2) Circle displacement: the initially generated circles are displaced to new locations to meet quality constraints including relations between regions maintenance and overlap avoidance. Because the initial circles can be easily generated in proportion to the statistical values, circular cartogram approaches are mainly developed to automate circle displacement. Dorling (1996) first proposed an approach to displace circles by simulating the orbits of stars and planets. Therefore, the circular cartogram is also called DorlingMap. But his approach has two shortcomings due to its iterative nature. First, it may be computationally challenging for large datasets. Tang (2013) then improved the efficiency of Dolring's approach by using graphic processing units. Second, Dorling's approach will inevitably result in local optimization, and the relations between regions may not well be maintained. Thus, some approaches aiming to achieve global optimization of involved constraints are also proposed. Inoue (Chrisman 1997) applied the least-squares adjustment for circular cartogram production. But his approach is hard to describe all involved constraints into unified linear functions, and the solution for these functions is also not guaranteed. Wei et al. (2022) introduced the elastic beam algorithm by using energy minimization principles, in which the circles are connected by a proximity graph (the elastic beam structure) according to the relations between circles. The circles that violate defined quality constraints are considered to have a force that deforms the beams. These deformations are modeled as energy, and the optimal shapes and locations of the beams are calculated under energy minimization principles iteratively to obtain new locations of circles. Their approach shows better performance by comparing to the above other approaches. Furthermore, many JavaScript libraries for data visualization such as D3 and Protovis also integrate circular cartograms due to their popularity (Bostock et al. 2011; Protovis – Dorling Cartograms 2010).

In summary, the commonly used Dorling's approach will result in local optimization. The elastic beam algorithm which can achieve global optimization of involved constraints is now proven to be better. But all existing approaches are developed for the visualization of static data, which is time-consuming and visual unstable between sequential views. It is very necessary to develop a new and efficient approach to produce stable multi-view circular cartograms for time-varying data visualization.

3 Quality requirements

Data visualization can be considered as an abstraction process to seek a legible but aesthetic form to represent information of interest with proper reality maintenance of the original data (Wei et al. 2022). These requirements can be developed into constraints to govern the visualization process. The quality constraints in circular cartogram production for time-varying data are summarized as follows:

1. *Legible constraints*: The circles in each circular cartogram should be distinguishable and no overlap is allowed (Dorling 1996).
2. *Maintenance constraints*

2.1 Topology relation (contiguity) maintenance: Topological relations between regions on the original map should be preserved as much as possible, specifically the adjacent relations between regions. This constraint is usually interpreted as contiguity maintenance, which means the contiguities between regions on the original map should be preserved as much as possible (Dorling 1996; Wei et al. 2022).

2.2 Relative relation maintenance: The relative relations between regions on the original map need to be preserved as much as possible, specifically the relative orientation relations (Chrisman 1997).
3. *Visual stability constraints*: It is desirable to minimize the changes between two sequential views as much as possible, and can be interpreted as the location deviation between corresponding circles in two sequential circular cartograms (Sondag et al. 2017).

The constraints summarized here are not like those defined in the database domain which need to be satisfied completely (Zhang et al. 2013). The maintenance constraints and visual stability constraints only need to be fulfilled as much as possible. But overlaps are not allowed in a circular cartogram and legible constraints must be fulfilled. Thus, the legible constraints (overlap avoidance) have the highest priority in our approach. Furthermore, many other kinds of constraints may also be defined based on specific user demands.

4 Methodology

4.1 Framework

Our framework is designed in Fig. 1. The example time-varying data are the population of each state in the USA in 1990, 2000, and 2010, Fig. 1a.

1. *Initial circle generation*: The initial circles are first generated with their locations in the centers of their corresponding regions and their sizes in proportion to the statistical values, Fig. 1(b). The relations between circles are represented as a proximity graph. This process can be easily implemented and is not illustrated in our approach. For more details, see (Dorling 1996).
2. *Hierarchy construction*: The time-varying data are grouped using a hierarchical clustering method based on the similarity between data at different time points, and a hierarchy is then built, the upper left part in Fig. 1c, Sect. 4.2.
3. *Circle displacement via hierarchical optimization*: The resultant locations of circles in each circular cartogram are obtained by using the improved elastic beam algorithm level by level according to the built hierarchy iteratively, Fig. 1c. In this iterative process, the intermediate circle locations in a cluster of similar circular cartograms at a lower level are generated from the intermediate circle locations at an upper level. The resultant circle locations in each cartogram are the final obtained locations for each cluster of circular cartograms at the lowest level. Because the elastic beam algorithm will be applied to displace circles in a cluster of similar circular cartograms simultaneously, the algorithm is improved in its proximity graph construction and force computation, Sect. 4.3. The hierarchy optimization for circle displacement by using the improved elastic beam algorithm is implemented in Sect. 4.4.

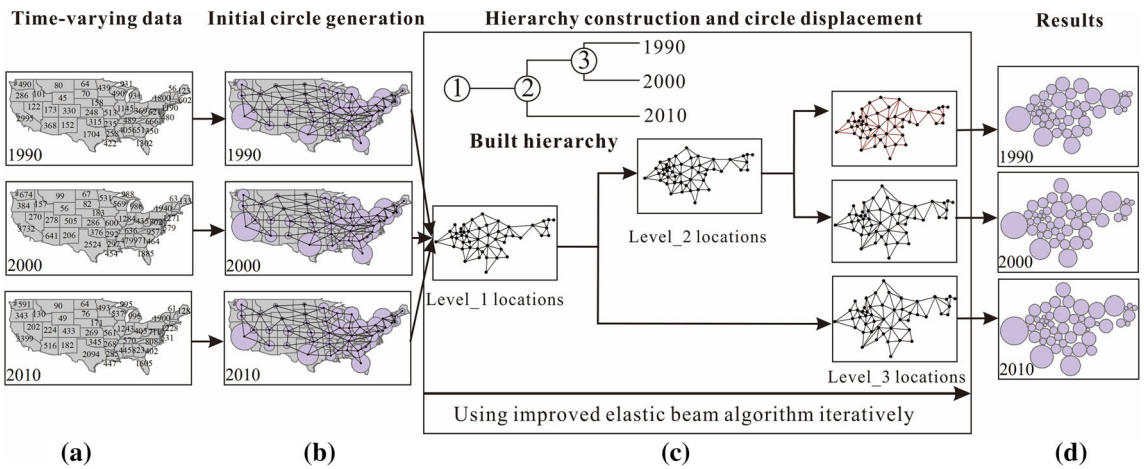

Fig. 1 The framework of the proposed approach. The example time-varying data are the population of each state in the USA (excluding Alaska and Hawaii) in 1990, 2000, and 2010

4.2 Hierarchy construction

The hierarchical clustering method is widely used for data clustering and a hierarchy can be built (Wei et al. 2018a). In this work, we also use the hierarchical clustering method to group the time-varying data. The clustering method starts with each element as a separate cluster and proceeds by merging the two closest clusters into a new cluster repeatedly until all elements are grouped into one cluster (Teichgraeber and Brandt 2018). Two measures need to be defined to implement the hierarchical clustering process: (1) distance between two elements, and (2) distance between two clusters.

Give a time-varying geographical statistical data: the geographical regions as $\{ regm\}_{m=1}^{M}$, where reg_m is a region; the time-varying statistical data as $\{ TVn\}_{n=1}^{N}$, $TVn = \{ v^n m\}_{m=1}^{M}$ is the statistical values for $\{ regm\}_{m=1}^{M}$ at a time point ($t = n$), and $v^n{}_m$ is the statistical value of reg_m at a time point ($t = n$). The distances between two elements and two clusters are defined as follows:

1. Distance between two elements

TV_n is an element in this work according to the previous definitions. Give two elements TV_x and TV_y, they represent the statistical values for the regions $\{ regm\}_{m=1}^{M}$ at time points ($t = x$ and $t = y$). The distance (De) between TV_x and TV_y can be measured by the value difference of their corresponding regions, as Eq. (1).

$$De(TVx, TVy) = \sum_{m=1}^{M} |v^x m - v^y m| \tag{1}$$

$v^x{}_m$ and $v^y{}_m$ are the statistical values of a region (reg_m) at time points ($t = x$ and $t = y$).

2. Distance between two clusters

$\{ TVx\}_{x=x'}^{X}$ is a cluster of elements in this work according to the previous definitions. Give two clusters of elements ($\{ TVx\}_{x=x'}^{X}$ and $\{ TVy\}_{y=y'}^{Y}$), they represent the statistical values for the regions $\{ regm\}_{m=1}^{M}$ from time points $t = x'$ to $t = X$ and $t = y'$ to $t = Y$. The distance (D_g) between $\{ TVx\}_{x=x'}^{X}$ and $\{ TVy\}_{y=y'}^{Y}$ can be defined based on the distance between all possible pairs (TV_x, TV_y) in the two clusters under different linkage criteria, such as single-linkage criterion, complete-linkage criterion, average-linkage criterion, etc. (Teichgraeber and Brandt 2018). Because we need to maintain the visual stability between sequential views, it means only the two clusters at adjacent time points can be grouped. Thus, the single-linkage criterion by using the shortest distance over all possible pairs (TV_x, TV_y) is suitable on this occasion. Furthermore, we set $Dg = \infty$ $(x' \neq Y \wedge y' \neq X)$ to guarantee that only the two clusters at adjacent time points can be grouped. Based on the above analysis, the Dg is defined as Eq. (2).

$$Dg(\{TVx\}_{x=x'}^{X}, \{TVy\}_{y=y'}^{Y}) = \begin{cases} \min\limits_{x,y} \|De(TVx, TVy)\| & (x' = Y \vee y' = X) \\ \infty & (else) \end{cases} \tag{2}$$

Based on the two defined measures, the hierarchical clustering method is then implemented by merging the two closest clusters into a new cluster repeatedly until all elements are grouped into one cluster. Then the hierarchy for the time-varying data is built and can be denoted as $\{ Clustersk\}_{k=1}^{N}$, $Clusters_k$ is the clustering result at level k. The example built hierarchy for time-varying data see the upper left part of Fig. 1c.

4.3 The improved elastic beam algorithm

In this work, the elastic beam algorithm will be applied to displace circles in a cluster of circular cartograms simultaneously. Thus, the existing elastic beam algorithm needs to be improved by considering the potential displacement of circles in a cluster of circular cartograms. The (1) proximity graph construction and (2) force computation are the two key processes of the elastic beam algorithm, and our improvements are also implemented in the two processes. For more details about the existing elastic beam algorithm, see (Wei et al. 2022).

Suppose a hierarchy $\{ Clustersk\}_{k=1}^{N}$ has been built for the time-varying statistical data $\{ TVn\}_{n=1}^{N}$ of the geographical regions $\{ regm\}_{m=1}^{M}$. Suppose a sub-cluster of time-varying statistical data at a level in the built hierarchy denoted as $\{ TVx\}_{x=x'}^{X}$; the intermediate circles representing $\{ TVx\}_{x=x'}^{X}$ of $\{ regm\}_{m=1}^{M}$ as

$\{ CSx\}_{x=x'}^{X}$, where $CSx = \{ C^x m\}_{m=1}^{M}$ are the circles representing the statistical data of $\{ regm\}_{m=1}^{M}$ at a time point ($t = x$), $C^x{}_m$ is the circle representing statistical data of reg_m at a time point ($t = x$).

1. Proximity graph construction

The proximity graph $G = (E, V)$ connecting all circles is considered as the elastic beam structure in the elastic beam algorithm, where $V = \{ nm\}_{m=1}^{M}$ is the node-set and n_m represents the circles $\{ C^x m\}_{x=x'}^{X}$; E is the edge set, and the edges in E mean that their connected circles will have forces in the elastic beam algorithm. The E is built by considering the constraints of topology relation maintenance and overlap avoidance (Sect. 3): (1) Edges for attractive forces: If two regions in $\{ regm\}_{m=1}^{M}$ have an adjacent relation, but their represented circles are not adjacent to each other, the topology relation maintenance constraint rules that the two separated circles need to attract each other to maintain the topology relations. Then an edge is inserted into E, which means that the two circles connected by the edge will have attractive forces. (2) Edges for repulsive forces: If two circles in $\{ C^x m\}_{m=1}^{M}$ at time points ($t = x'$ to $t = X$) have an overlap, the overlap avoidance constraint rules that the two circles need to repulse each other to avoid the overlap. Then an edge is inserted into E, which means that the two circles connected by the edge have repulsive forces.

The proximity graph is constructed for the displacement of the intermediate circles $\{ CSx\}_{x=x'}^{X}$ representing statistical data at time points ($t = x'$ to $t = X$) as Algorithm 1.

Algorithm 1. Proximity graph construction

Input: the regions as $\{reg_m\}_{m=1}^{M}$, the intermediate circles representing statistical data at time points ($t = x'$ to $t = X$) as $\{CS_x\}_{x=x'}^{X}$, in which the intermediate circles at the time point($t = x$) is $CS_x = \{C^x{}_m\}_{m=1}^{M}$

Output: the proximity graph $G = (E, V)$

Initialize: the proximity graph as $G = (E, V)$, $V = \{n_m\}_{m=1}^{M}$, and n_m is computed as the center of the circles; and $E = Null$

For each node pair (n_i, n_j), $n_i \in V$, $n_j \in V$ **Do**

  If reg_i and reg_j have an adjacent relation AND $e(n_i, n_j) \notin E$

    Then add $e(n_i, n_j)$ to E

  For each $CS_x = \{C^x{}_m\}_{m=1}^{M}$ in $\{CS_x\}_{x=x'}^{X}$ **Do**

    If $C^x{}_i$ and $C^x{}_j$ have an overlap AND $e(n_i, n_j) \notin E$

      Then add $e(n_i, n_j)$ to E

Return $G = (E, V)$

2. Force computation

Two kinds of forces need to be computed as illustrated in the proximity graph construction: (1) If two circles connected by an edge in G have an overlap, they will accept repulsive forces to repulse each other to avoid overlap, and (2) if two circles connected by an edge in G are separated, they will accept attractive forces to attract each other to maintain the topology relations. Suppose a proximity graph (G) is constructed for the intermediate circles $\{ CSx\}_{x=x'}^{X}$ representing statistical data at time points ($t = x'$ to $t = X$). Then given an edge $e(n_i, n_j)$ in G, its connected circle pairs are denoted as $\{ (C^x i, C^x j)\}_{x=x'}^{X}$, where $(C^x i, C^x j)$ is a circle pair in a circular cartogram at a time point ($t = x$). The two kinds of forces need to be computed by considering all circle pairs in $\{ (C^x i, C^x j)\}_{x=x'}^{X}$.

For each pair $(C^x i, C^x j)$ connected by the edge $e(n_i, n_j)$, the repulsive forces ($ref^x{}_i$ and $ref^x{}_j$) and attractive forces ($atf^x{}_i$ and $atf^x{}_j$) on the two nodes (n_i and n_j) can be computed according to Wei et al. (2022) as Eqs. (3)–(6), Fig. 2. Because we try to generate an intermediate result for statistical data at time points ($t = x'$ to $t = X$). The final repulsive forces (f_ref_i and f_ref_j) and attractive forces (f_atf_i and f_atf_j) on the two nodes (n_i and n_j) are computed as an average of the obtained $ref^x{}_i$, $ref^x{}_j$, $atf^x{}_i$ and $atf^x{}_j$ for each circle pair $(C^x i, C^x j)$ in $\{ (C^x i, C^x j)\}_{x=x'}^{X}$ as Eqs. (3)–(6).

$$f\_refi = \frac{\sum_{x=x'}^{X} ref^x i}{X - x'} \quad \left(ref^x i = \frac{\overrightarrow{njni}}{\left|\overrightarrow{ninj}\right|} * \frac{R^x j}{R^x i + R^x j} * |L^x| \right) \tag{3}$$

$$f\_refj = \frac{\sum_{x=x'}^{X} ref^x j}{X - x'} \quad \left(ref^x j = \frac{\overrightarrow{ninj}}{\left|\overrightarrow{ninj}\right|} * \frac{R^z i}{R^z i + R^z j} * |L^x| \right) \tag{4}$$

$$f\_atfi = \frac{\sum_{x=x'}^{X} atf^x i}{X - x'} \quad \left(atf^x i = \frac{\overrightarrow{ninj}}{\left|\overrightarrow{ninj}\right|} * \frac{R^x j}{R^x i + R^x j} * L^x \right) \tag{5}$$

$$f\_atfj = \frac{\sum_{x=x'}^{X} atf^x j}{X - x'} \quad \left(atf^x j = \frac{\overrightarrow{njni}}{\left|\overrightarrow{ninj}\right|} * \frac{R^x i}{R^x i + R^x j} * L^x \right) \tag{6}$$

where $R^x{}_i$ and $R^x{}_j$ are the radii of $C^x{}_i$ and $C^x{}_j$, L^x is the distance between the circle pair ($C^x{}_i$ and $C^x{}_j$), and L^x is defined as $L^x = Dis(ni, nj) - (R^x i + R^x j)$, and $Dis(n_i, n_j)$ is the distance between two nodes n_i and n_j.

After the forces for all nodes are computed based on each edge in G, a node may be affected by more than one individual external force. We then combine these forces on a node into a resultant one by combining the local maximum component forces in four main directions according to Wei et al. (2022). Based on the built proximity graph and computed forces for each node, the elastic beam algorithm is then applied to the proximity graph to find new intermediate locations of nodes (namely the circles).

4.4 Hierarchical optimization

The improved elastic beam algorithm is applied iteratively to generate different levels of intermediate results level by level according to the built hierarchy. Two kinds of stop conditions need to be defined: (1) when to stop the iterative process for the generation of an intermediate result at a specific level in the built hierarchy; (2) when to stop the whole iterative process.

1. The stop condition for each generation of an intermediate result at a specific level can be defined by setting a threshold for the iterative process, as *iTs*. It means the iterative process to generate an intermediate result for the statistical data at a specific level in the built hierarchy will stop if the iterative step reaches *iTs*, then another iterative process starts based on the intermediate result to generate a new intermediate result for the statistical data at a lower level.

2. The stop condition for the whole iterative process can be defined based on whether a satisfactory result is obtained. If a satisfactory result is obtained, the circles in the circular cartograms don't need to be displaced. This can be ruled by setting a threshold for the maximum forces ($\max(f_k)$) for all circles in the last application of the elastic beam algorithm (Wei et al. 2022). If $\max(fk) \leq \varepsilon$ and ε=0.001, then the whole iterative process to displace circles in a circular cartogram for a specific time point stops.

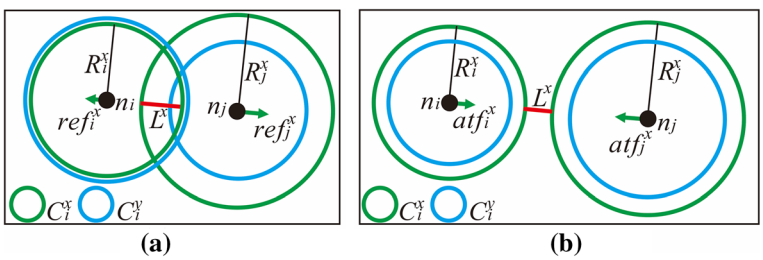

Fig. 2 Force computation for nodes n_i and n_j which are connected by an edge $e(n_i, n_j)$. The figure is an example in which two circle pairs ($c^x{}_i$,$c^x{}_j$) and ($c^y{}_i$,$c^y{}_j$) are considered, this figure is redrawn according to (Wei et al. 2022). **a** Repulsive forces; **b** attractive forces

Furthermore, not all topology relations between regions can be maintained as pointed out in Wei et al. (2022), it means some attractive forces are redundant. Because circles will be attracted to be close to each other as much as possible with the attractive forces after a maximum step (*aTs*) is reached. To this end, if the *aTs* is reached and a satisfactory result cannot be obtained for a circular cartogram, the attractive forces are not considered in later iterative steps. The iterative process for hierarchical optimization by using the elastic beam algorithm is implemented as Algorithm 2.

Algorithm 2. Iterative process for hierarchical optimization

Input: the regions as $\{reg_m\}_{m=1}^{M}$, the time-varying statistical data for these regions as $\{TV_n\}_{n=1}^{N}$, the maximum step to generate an intermediate result for the statistical data at a specific level in the built hierarchy is *iTs*, the maximum step without considering the attractive forces is *aTs*

Initialize: the iterative step to generate a result for the data at different times as $s \leftarrow 0$

Construct the hierarchy for $\{TV_n\}_{n=1}^{N}$ as $\{Clusters_k\}_{k=1}^{N}$, $Clusters_k$ is the clustering result at level k

Generate initial circles for $\{TV_n\}_{n=1}^{N}$ as $\{CS_x\}_{x=1}^{N}$, $CS_x = \{C^x{}_m\}_{m=1}^{M}$ are the circles representing the statistical data of $\{reg_m\}_{m=1}^{M}$ at a time point ($t = x$)

For i $\leftarrow$ 1 to N **Do**

Get the i-th level clusters in $\{Clusters_k\}_{k=1}^{N}$ as $Clusters_i$

For each sub-cluster in $Cluster_i$

Set the iterative step for the generation of an intermediate result as p, and $p \leftarrow 0$

Do

Get the corresponding circles for the data in the sub-cluster as $\{CS_x\}_{x=x'}^{X}$; construct the proximity graph (G) for $\{CS_x\}_{x=x'}^{X}$;

If $s \le aTs$ **Then** force computation with attractive forces

Else Then force computation without attractive forces

Compute the displacement vector for each node in G with the elastic beam algorithm, the maximum force is $\max(f_k)$;

Update the coordinates of the circles in $\{CS_x\}_{x=x'}^{X}$ based on the displacement vector

$p \leftarrow p + 1$

While ($\max(f_k) \le \varepsilon$ AND $p \le iTs$)

$s \leftarrow s + p$

End

Return $\{CS_x\}_{x=1}^{N}$

5 Experiment

5.1 Experiment setup

1. Dataset

Two widely used datasets for circular cartogram production are adopted in our approach. Dataset A: the population of the USA excluding Alaska and Hawaii from 1980 to 2015, the time interval is 5 years (8-time points), and the data source: https://www.ipums.org/. Dataset B: the population of Europe countries excluding Iceland from 1985 to 2020, the time interval is 5 years (8-time points), and the data source: https://data.worldbank.org/indicator/SP.POP.GROW?locations=EU.

2. Parameter setting

The circle sizes for Dataset A are defined as proportional to the data values as Eq. (7) (Dorling 1996). While the circle sizes for Dataset B are defined as Eq. (8) (Wei et al. 2022) because the data values in Dataset B present a huge variation which will lead to huge differences in the circle sizes if the circle sizes are still defined based on Eq. (7).

$$R_i = \sqrt{vi/v_{\min}} * R_{\min} \tag{7}$$

$$R_i = \sqrt{(vi - v_{\min})/(v_{\max} - v_{\min})} * (R_{\max} - R_{\min}) + R_{\min} \tag{8}$$

where $v_{\max}$ and $v_{\min}$ are the maximum and minimum data values. R_i is the radius of a circle that represents a region (reg_i) with data value v_i. $R_{\min}$ is the minimum radius of the circles and $R_{\min}$ = 1 mm (Liu et al. 2014). $R_{\max}$ is the maximum radius of the circles and $R_{\max}$ = 20 mm.

Two parameters (*iTs* and *aTs*) defined in Sect. 4.4 also need to be set in our approach. According to the parameter settings in Wei et al. (2022), the two parameters are set by considering the number of the circles (*NumC*) and the time points (*TC*) as: $iTs = \lfloor 4 * NumC/TC \rfloor$, $iTs = 2 * NumC$. If $NumC < 10$, then $NumC$=10; if $NumC > 50$, then $NumC$=50.

3. Experimental environment

We use C# code to implement our proposed approach based on ArcEngine 10.2. The experiments are carried out on a personal computer (Intel® Core™ 1.60 GHz i5-8265U CPU and 8G RAM).

4. Results

The circular cartograms for the two datasets are generated by using the proposed approach and are shown in calendar-like forms, Figs. 3 and 4. The timeline forms are also available in our uploaded files "Fig. 3-Gif for the USA" and "Fig. 4-Gif for Europe". Evaluation and comparison are implemented in Sect. 5.2.

5.2 Evaluation and comparison

1. The existing approach for comparison

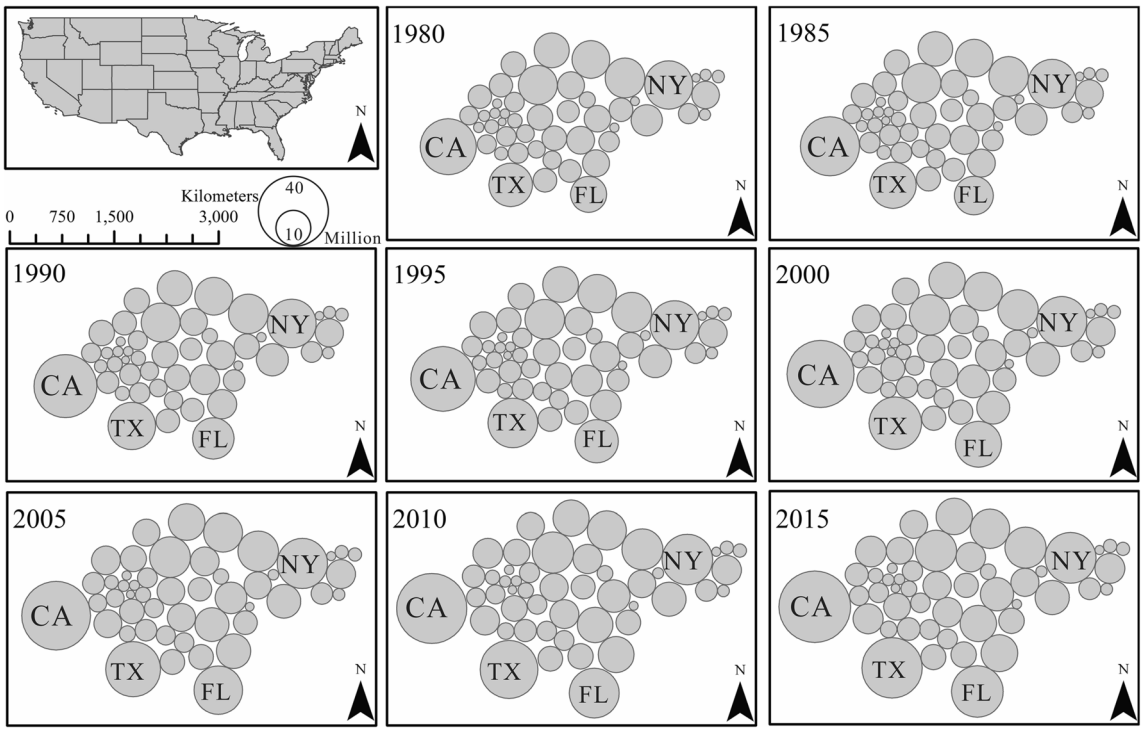

Fig. 3 The circular cartogram generated by using the proposed approach for Dataset A

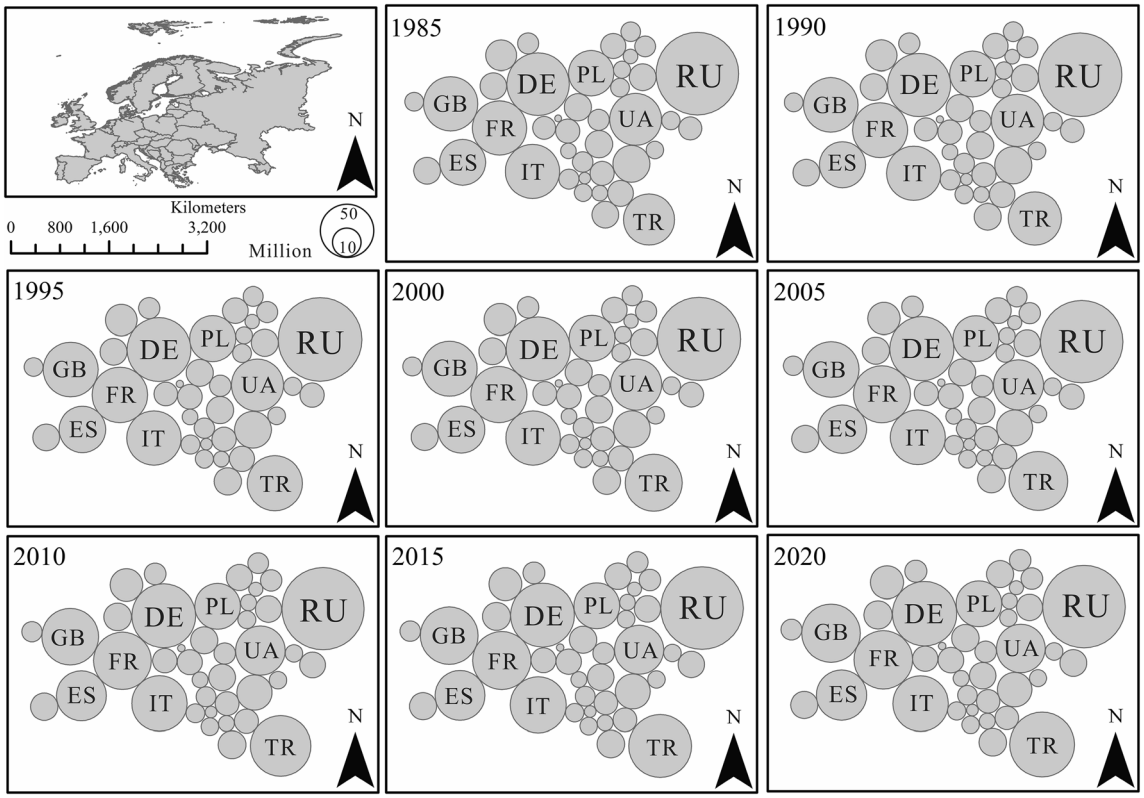

Fig. 4 The circular cartogram generated by using the proposed approach for Dataset B

The circular cartograms for comparisons are generated based on the same two datasets by using the existing approach (Wei et al. 2022). Unlike the proposed approach, the existing approach computes each circular cartogram for different time points separately by using the existing elastic beam algorithm with the same parameter settings.

2. Effectiveness evaluation and comparison

The effectiveness is evaluated by using the following measures according to the quality requirements in Sect. 3: (1) The average number of overlaps in each circular cartogram (*Ave_NumO*) to evaluate the avoidance of overlaps; (2) The average ratio to maintain the adjacent relations in each circular cartogram (*Ave_RT*) to evaluate the maintenance of topology relations; (3) The average root mean square which describes the variation of direction angles for the links between the centers of neighboring circles (*Ave_RMS*) in each circular cartograms to evaluate the maintenance of relative relations (Chrisman 1997); (4) The average total location deviation of the corresponding circles in two sequential circular cartograms (*Ave_TDD*) to evaluate visual stability, and *Ave_TDD* is computed based on geographic distance. The statistical results on the effectiveness of the proposed approach and the existing approach are shown in Table 1.

Table 1 Statistical results on the effectiveness and efficiency of the proposed approach and the existing approach

	Proposed approach		Existing approach	
	Dataset A	Dataset B	Dataset A	Dataset B
Ave_NumO↓	0	0	0	0
Ave_RT↑ (%)	68.12	65.33	**69.96**	65.33
Ave_RMS(°)↓	35.47	**31.30**	**35.19**	31.48
Ave_TDD (10^6 m)↓	**10.89**	**4.11**	31.02	15.03
IS↓	**571**	**408**	1068	907
t(s)↓	**255.15**	**180.27**	404.52	313.60

From Table 1, we have the following observations. (1) No overlaps are generated by using the two approaches. These indicate that the two approaches can both avoid overlaps. (2) Comparing the proposed approach to the existing approach for Dataset A: the *Ave_RT* will reduce by 1.84%, the *Ave_RMS* will increase by 0.28 °, while the *Ave_TDD* will reduce by 20.13 $*10^6$ m and the reducing rate is 184.85%. (3) Comparing the proposed approach to the existing approach for Dataset B, the *Ave_RT* will stay the same, the *Ave_RMS* will reduce by 0.18 °, while the *Ave_TDD* will reduce by 10.92 $*10^6$ m and the reducing rate is 265.69%. The observations (2) and (3) both indicate that the *Ave_TDD* can be far better achieved and the *Ave_RT* and *Ave_RMS* can be similarly achieved by using the proposed approach, which means better visual stability and similar quality on overlap avoidance and relation maintenance are achieved by using the proposed approach in comparison with the existing approach.

3. Efficiency evaluation and comparison

The efficiency is evaluated using time consumption (*t*). Furthermore, the iterative step (*IS*) of using the elastic beam algorithm is another key factor that will influence the efficiency and is also evaluated. The statistical results on the efficiency of using the proposed approach and the existing approach are shown in Table 1.

From Table 1, we have the following observations. (1) The iterative steps for the proposed approach are 571 (Dataset A) and 408 (Dataset B), and 1068 (Dataset A) and 907 (Dataset B) for the existing approach, which indicates the *IS* can be improved by 87.04% (Dataset A) and 122.30% (Dataset B) with the proposed approach. (2) The time consumptions are 255.15 s (Dataset A) and 180.27 s(Dataset B) for the proposed approach, and 404.52 s (Dataset A) and 313.60 s(Dataset B) for the existing approach, which indicates the *t* can be improved by 58.54% (Dataset A) and 73.96%(Dataset B) with the proposed approach. The two observations both indicate that the proposed approach is more efficient than the existing approach.

5.3 User study

To validate the effectiveness of the visual stability in multi-view circular cartograms, we developed a game named "Circle matching."

1. Design

Visual stability is a constraint to keep the visual consistency of corresponding objects at sequential views (Chi et al. 2015; Sondag et al. 2017). Good visual stability can enhance the ability of a user to search for an object in a new view after the user notices its corresponding object in a previous view. Our game is designed based on the hypothesis, Fig. 5.

The game interface has two views which display the two circular cartograms at sequential times. A region name will be displayed in the middle textbox, e.g., NY. The users are asked to find NY in the left view first. A click on the circle labeled NY is the sign that the target circle NY is found, and the circle will be highlighted only when the correct circle is clicked. Then the users are asked to find NY in the right view as soon as possible. The moments of the two correct clicks in the two views are recorded, and the searching

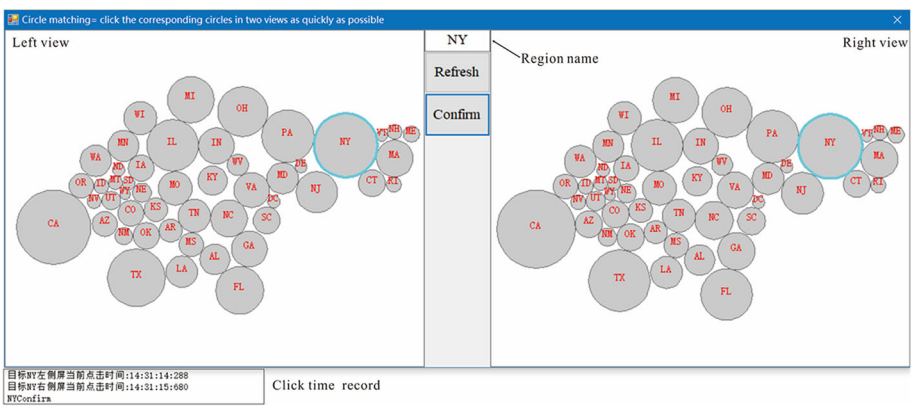

Fig. 5 The interface of the designed game for the user study

time can then be computed. The refresh button is a button to switch to the next two new circular cartograms at sequential times, and a new region name is also given.

We have embedded 14 pairs of circular cartograms at sequential times along with 14 region names respectively which are generated by the proposed approach and the existing approach in the game. The 14 pairs cover the period in Fig. 3, and the circles representing 14 regions also have different sizes.

2. Participant

4 students or staff (2 male and 2 female) from the Aerospace Information Research Institute, Chinese Academy of Sciences who have experience in visualization took the experiments. Hence, participants can be considered reliable experimentees.

3. Procedure

The participants have been informed of the rules of the game first. Then the participants are asked to take a pre-test that contains 10 pairs of circular cartograms at sequential times along with 10 region names first. After a confirmation that the participants understand the rules and process, our user study is implemented on the embedded 14 pairs of circular cartograms, respectively, which are generated by the proposed approach and the existing approach.

4. Result

We recorded the searching time (*St*) for the same region names in 14 pairs of circular cartograms generated by the proposed approach and the existing approach, respectively. The minimum searching time (*Min_St*), maximum searching time (*Max_St*), and average searching time(*Ave_St*) are computed, and results are shown in Table 2. As shown in Table 2, we can see that the *Ave_St* of User_1, User_2 and User_3 is smaller, and the *Ave_St* of User_4 is nearly the same while comparing the proposed approach to the existing approach. These indicate that visual stability can enhance the ability of a user to search for an object in a new view after the user notices its corresponding object in a previous view.

6 Discussion

In this section, we analyze the influences on different strategies used in hierarchy construction, proximity graph construction, and force computation.

1. Hierarchy construction

We define the distance between two clusters (*Dg*) by using the shortest distance over all possible pairs in the two clusters based on the single-linkage criterion in hierarchy construction, Sect. 4.2. The complete-linkage criterion or the average-linkage criterion can also be applied to define the distance between two clusters by using the longest distance or average distance over all possible pairs in the two clusters as *C_Dg* and *A_Dg*. We also generate the circular cartograms in which *C_Dg* and *A_Dg* are adopted to build the hierarchy, and the statistical results on the effectiveness and efficiency are shown in Table 3.

As shown in Table 3, we can see that the *Ave_NumO*, *Ave_RT*, *Ave_RMS*, *Ave_TDD*, *IS* are the same, and time consumption (*t*) is almost the same while using *C_Dg* or *A_Dg*. It is because the same hierarchies are built by using *C_Dg* or *A_Dg*. Comparing Tables 1 and 3, we have the following observations: 1) The *Ave_NumO* is the same by comparing the *Dg* to *C_Dg* and *A_Dg*; these indicate that they can all well avoid overlaps. 2) The *Ave_RT* increases by 1.49%, *Ave_RMS* reduces by 0.34 ° by comparing the *Dg* to *C_Dg* and *A_Dg*; these indicate that *C_Dg* and *A_Dg* can a little better maintain the relations between regions. 3) The *Ave_TDD* increases $0.17*10^6$ m, *IS* increases 6, and time consumption(*t*) increases 7.06 s and 4.98 s by

Table 2 Statistical results on the user study

	Proposed approach			Existing approach		
	Min_St	*Max_St*	*Ave_St*	*Min_St*	*Max_St*	*Ave_St*
User_1	0.307	1.341	**0.956**	0.663	1.461	1.108
User_2	0.546	1.263	**0.894**	0.658	1.430	0.985
User_3	0.263	1.870	**1.046**	0.41	1.890	1.129
User_4	0.648	2.043	1.228	0.516	2.349	**1.211**

Table 3 Statistical results on the generated circular cartograms by using *Dg*, *C_Dg*, and *A_Dg* in hierarchy construction

	Dg	*C_Dg*	*A_Dg*
Ave_NumO↓	0	0	0
Ave_RT↑ (%)	68.12	**69.61**	**69.61**
Ave_RMS(°)↓	35.47	**35.13**	**35.13**
Ave_TDD (10^6 m)↓	**10.89**	11.06	11.06
IS↓	**571**	577	577
t(s) ↓	**255.15**	262.21	260.13

comparing the *Dg* to *C_Dg* and *A_Dg*; these indicate that *Dg* can a little better maintain the visual stability and is also a little more efficient. Based on the above observations, we can conclude that the three criteria used to define the distance between two clusters may generate similar results, and the *Dg* is recommended if visual stability and efficiency are considered in higher priority, and *C_Dg* and *A_Dg* are recommended if relation maintenance is considered in higher priority.

2. Proximity graph construction

The proximity graph is constructed by considering the adjacent relations between regions and overlaps between circles in our proposed approach, Sect. 4.3. However, many other kinds of proximity graphs can also be applied based on different user demands, such as Gabriel graph (GG), relative neighbor graph (RNG), minimum spanning tree (MST), or nearest neighbor graph (NNG) (Guo et al. 2017).To this end, we also generate circular cartograms by using the widely used MST in proximity graph construction. Furthermore, edges are also inserted into the MST if two circles overlap because the overlaps must be avoided in a circular cartogram. The result is shown in Fig. 6. As shown in Fig. 6, the circular cartograms with no overlaps can be successfully generated. The relations defined based on the MST are also maintained, and these circles result in an MST-like arrangement. The MST can also be changed into any other defined proximity graph (*G*) by users, and *G*-like arrangements can then be generated. These results indicate that the proposed approach can be applied for arbitrary proximity graphs, and this conclusion has also been proved in Wei et al. (2022).

3. Force computation

The average force (*Ave_F*) is adopted in force computation Sect. 4.3, as defined in Eqs. (3)–(6). The maximum force (*MaxF*) and minimum force (*MinF*) can also be adopted. We have also generated the circular cartograms with *MaxF* or *MinF* in force computation. Because we found the iterative process won't

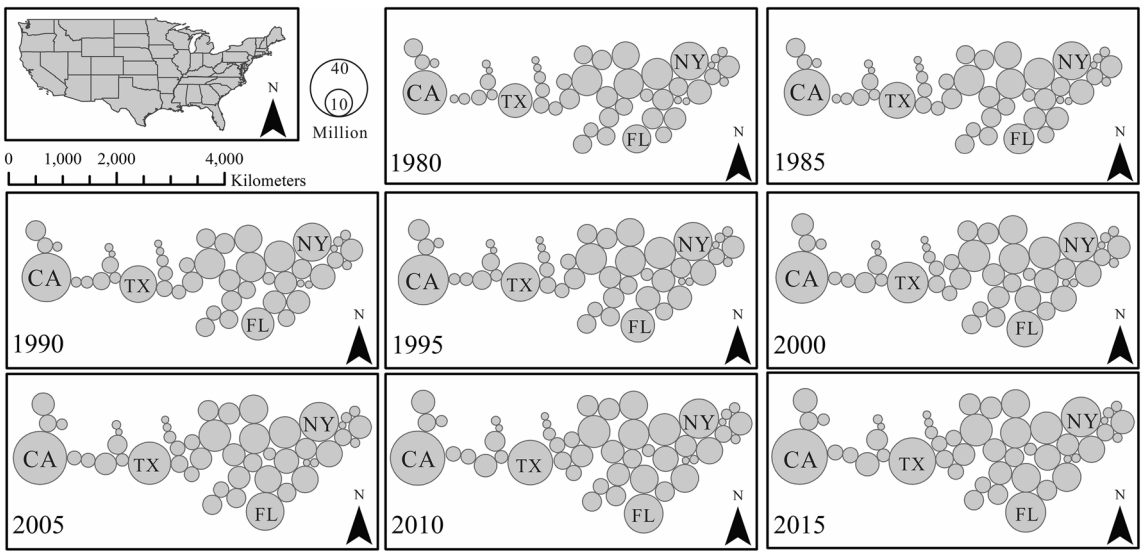

Fig. 6 The circular cartogram generated using the proposed approach in which the proximity graph is built based on MST for Dataset A

Table 4 Statistical results on the generated circular cartograms by using *MaxF* and *MinF* in force computation

	Ave_F	*MaxF*	*MinF*
Ave_NumO↓	0	0	28.75
Ave_RT↑ (%)	68.12	67.31	–
Ave_RMS(°)↓	35.47	35.80	–
Ave_TDD (10^6 m)↓	10.89	18.52	–
IS↓	571	582	–
t(s) ↓	255.15	258.69	–

stop if *MinF* is adopted, a maximum step ($Ts = 4 * NumC$) is set for the iterative process of using the improved elastic beam algorithm for each circular cartogram while *MinF* is adopted. The statistical results on the effectiveness and efficiency are shown in Table 4.

From Table 4, we have the following observations: (1) The *Ave_RT* will reduce by 0.81%, the *Ave_RMS* will increase by 0.33 °, the *Ave_TDD* will increase by $7.63*10^6$ m, the *IS* will increase by 11, the time consumption(*t*) will increase 3.54 s, and the *Ave_NumO* is 0 if the *MaxF* is adopted; these indicate the average force (*Ave_F*) adopted in our approach is better in relation maintenance and visual stability is more efficient, and can both avoid overlaps by comparing the *MaxF*. (2) If the *MinF* is adopted, satisfactory results are hard to be obtained because the iterative process is hard to stop; and the *Ave_NumO* is 28.75 if a setting *Ts* is reached, it indicates that the overlaps can't be avoided with *MinF*.

4. Others

Although satisfactory results can be generated by using the proposed approach with defined quality constraints in Sect. 3. We also proved that an arbitrary proximity graph can be built for different considering quality constraints and satisfactory circular cartograms can be obtained. But the constraints may vary according to different user demands, e.g., some users may focus on the differences between sequential view and visual stability may not be considered on this occasion. Thus, our approach may still need some adjustments by considering these specific user demands.

The size and shape are the two main graphic variables that are applied in the proposed approach. The color may be another important graphic variable that can be used and may need to be explored in future works. For example, different colors can be assigned to different administrative regions or circles to enable a more distinguished view (Wei et al. 2018b). Furthermore, the colors can also be used to represent a property of the regions, such as the changing rate over time.

7 Conclusion

To produce visually stable circular cartograms for time-varying data efficiently, we generate circular cartograms at different time points all in once. The circles' locations are obtained level by level according to a built hierarchy, in which the hierarchy is built based on the similarity between data at different times with a hierarchical clustering method. The circle locations in more similar circular cartograms are computed based on a more recent intermediate result. The potential displacement of circles in a cluster of similar circular cartograms is implemented by improving the elastic beam algorithm in its proximity graph construction and force computation. Experiments demonstrate that the proposed approach can achieve higher quality on visual stability and higher efficiency with almost the same quality on overlap avoidance and relation maintenance by comparing the existing approaches. Furthermore, our proposed approach is also applicable for arbitrary built proximity graphs by considering varied user demands.

Future works will focus on: (1) Apply more graphic variables such as colors to create more informed visualization; (2) Improve the algorithm efficiency.

Acknowledgements The authors wish to thank Dr. Tingzhong Huang for his help in data collecting. This work was supported in part by a grant from National Natural Science Foundation of China, Grant Number [41871378].

References

Bostock M, Ogievetsky V, Heer J (2011) D3 data-driven documents. IEEE Trans vis Comput Graphics 17(12):2301–2309

Buchin K, Speckmann B, Verdonschot S (2012) Evolution strategies for optimizing rectangular cartograms. In: International conference on geographic information science. Springer, Berlin, pp 29–42
Castells M, Blackwell C (1998) The information age: economy, society and culture: the rise of the network society. Environ Plann B Plann Des 25(1):631–636
Chi MT, Lin SS, Chen SY, Lin CH, Lee TY (2015) Morphable word clouds for time-varying text data visualization. IEEE Trans Visual Comput Graphics 21(12):1415–1426
Chrisman NR (1997) Cartogram projections of planar polygon networks. Harvard Laboratory for Computer Graphics and Spatial Analysis, Harvard University
Dorling DFL (1996) Area cartograms: their use and creation. In: Concepts and techniques in modern geography series, Environmental Publications, University of East Anglia
Dougenik JA, Chrisman NR, Niemeyer DR (1985) An algorithm to construct continuous area cartograms. Prof. Geographer 37(1):75–81
Gastner MT, Newman MEJ (2004) Diffusion-based method for producing density-equalizing maps. Proc Natl Acad Sci 101(20):7499–7504
Gastner MT, Seguy V, More P (2018) Fast flow-based algorithm for creating density-equalizing map projections. Proc Natl Acad Sci 115(10):E2156–E2164
Guo Q, Wei Z, Wang Y, Wang L (2017) The method of extracting spatial distribution characteristics of buildings combined with feature classification and proximity graph. Acta Geodaetica Et Cartogr Sin 46(5):631–638
Heilmann R, Keim D, Panse C, Sips M (2004) Recmap: rectangular map approximations. In: IEEE symposium on information visualization, IEEE, pp.33–40
Inoue R (2011) A new construction method for circle cartograms. Cartogr Geogr Inf Sci 38(2):146–152
Jackel CB (1997) Using ArcView to create contiguous and noncontiguous area cartograms. Cartogr Geogr Inf Syst 24(2):101–109
Kronenfeld BJ (2018) Manual construction of continuous cartograms through mesh transformation. Cartogr Geogr Inf Sci 45(1):76–94
Liu Y, Guo Q, Sun Y, Ma X (2014) A combined approach to cartographic displacement for buildings based on skeleton and improved elastic beam algorithm. PLoS ONE 9(12):e113953
Nusrat S, Kobourov S (2016) The state of the art in cartograms. Comput Graphics Forum 2(3):619–642
Protovis – Dorling Cartograms (2010). http://mbostock.github.io/protovis/ex/cartogram.html
Raisz E (1934) The rectangular statistical cartogram. Geogr Rev 24(3):292–296
Reyes Nunez JJ (2014) The use of cartograms in school cartography. Thematic Cartography for the Society, 327–339
Sondag M, Speckmann B, Verbeek K (2017) Stable treemaps via local moves. IEEE Trans Visual Comput Graphics 24(1):729–738
Speckmann B, Kreveld M, Florisson S (2006) A linear programming approach to rectangular cartograms. In: International symposium on spatial data handling (SDH'06)
Sun S (2013a) A fast free-form rubber-sheet algorithm for contiguous area cartograms. Int J Geogr Inf Sci 27(3):567–593
Sun S (2013b) An optimized rubber-sheet algorithm for continuous area cartograms. Prof Geogr 65(1):16–30
Sun S (2020) Applying forces to generate cartograms: a fast and flexible transformation framework. Cartogr Geogr Inf Sci 47(5):381–399
Tang W (2013) Parallel construction of large circular cartograms using graphics processing units. Int J Geogr Inf Sci 27(11):2182–2206
Teichgraeber H, Brandt AR (2018) Systematic comparison of aggregation methods for input data time series aggregation of energy systems optimization problems. Comput Aided Chem Eng 44:955–960
Van Kreveld M, Speckmann B (2007) On rectangular cartograms. Comput Geom 37(3): 175–187
Wei Z, Guo Q, Wang L, Yan F (2018a) On the spatial distribution of buildings for map generalization. Cartogr Geogr Inf Sci 45(6):539–555
Wei Z, Guo Q, Yan F, Wang Y (2018b) Backtracking method of coloring administrative maps considering visual perception rules. Acta Geodaetica Et Cartogr Sin 47(3):396–402
Wei Z, Ding S, Xu W, Cheng L, Zhang S, Wang Y (2022) Circular cartograms via the elastic beam algorithm originated from cartographic generalization. https://arxiv.org/abs/2204.12645
Wolf EB (2005) Creating contiguous cartograms in ArcGIS 9. In: Proceedings of 2005 ESRI international user conference, San Diego, CA
Ying S, Dou X, Xu Y (2021) Visualization of the epidemic situation of COVID-19. J Geo-Information Sci 23(2):211–221
Zhang X, Stoter J, Ai T, Kraak MJ, Molenaar M (2013) Automated evaluation of building alignments in generalized maps. Int J Geogr Inf Sci 27(8):1550–1571