

# 基于区间分类的螺旋图可视化边绑定方法

朱立霞 李天瑞 滕飞 彭博

(西南交通大学信息科学与技术学院 成都 611756)

**摘要** 在时序数据可视化领域,螺旋图是一种常用的可视化方法,它既能将多个阶段的数据同时展示在一个平面空间内,又能在有限的空间内展示任意时长的数据。针对现有的螺旋图可视化方法在展示大量的时间序列数据时会出现因螺旋线交叉而导致视觉杂乱的问题,研究螺旋图可视化方法意义非凡。首先将状态圆环上的数据点进行分类;然后在相邻的状态圆环之间设置虚拟绑定圆环,通过边绑定的函数将状态圆环上的数据点映射到其对应的虚拟绑定圆环上;最后在状态圆环与其对应的虚拟绑定圆环之间绘制 Bézier 曲线,在虚拟绑定圆环与虚拟绑定圆环之间绘制螺旋线,从而实现边绑定的效果。实验结果表明,该边绑定算法能够有效地对大规模数据进行可视化,并能有效地缓解视觉杂乱的问题。

**关键词** 可视化,螺旋图,边绑定,时间序列

中图分类号 TP311.11 文献标识码 A DOI 10.11896/j.issn.1002-137X.2019.01.016

## Edge Bundling Method of Spiral Graph Based on Interval Classification

ZHU Li-xia LI Tian-rui TENG Fei PENG Bo

(School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, China)

**Abstract** Spiral graph is a common visualization method in visualizing time series data. It can not only simultaneous display the multiple-stages data in one plane space, but also demonstrate the data with different time length in a limited space. In order to solve the problem of visual clutter caused by the intersection of helical lines in the present spiral image visualization methods, a method of edge bundling is of great significance. First, the data points on the state circle are classified. Then the virtual bundling circles are set between the adjacent state circles, and the data points on the state ring are mapped to the corresponding virtual bundling circle by the function of edge bundling. Finally, in order to achieve the effect of curve bundling, the Bézier curve is drawn between the state circle and its corresponding virtual bundling circle, and the spiral curve is drawn between the virtual bundling circle and the virtual bundling circle. Experimental results show that the edge-bundling algorithm is effective for large-scale data visualization and can effectively alleviate the problem of visual clutter.

**Keywords** Visualization, Spiral graph, Edge bundling, Time series

## 1 引言

在互联网时代,数据产生和传播的速度越来越快,数据形式也越来越多。在大数据时代,时间序列数据便是一种在人们生产和生活中常见的数据类型,它包含多个状态的转换,每两个状态的转换之间都有多条数据,每条数据都有时间属性。很多研究人员希望从时间序列数据中获得数据随着时间变化的模式,但是通过肉眼观察很难从中快速获得有效的信息和知识。

近年来,可视化技术发展迅速。可视化技术旨在通过将数据抽象为图表来帮助人们理解晦涩难懂的数据,并通过交

互操作帮助人们发现隐藏在数据中的知识。在时间序列数据的可视化展示领域,人们往往青睐于曲线图、主题河流、树图和螺旋图等。其中,螺旋图是一种非常节省可视化页面,且能很好地展示时间序列数据的可视化方法。它使用圆环表示状态,采用一组同心圆环表示一个事件的多个状态,采用螺旋线表示相邻状态间的可达路径,螺旋线的起始位置由其起始时间与起始状态决定,终止位置由其结束时间与结束状态决定。

但是,现有的螺旋图可视化方法在展示大量数据时,会出现因螺旋线交叉而导致视觉杂乱的问题。基于此,本文提出了一种将处于同一时间区间的数据绘制成束的螺旋图边绑定方法——基于区间分类的螺旋图可视化边绑定方法。

到稿日期:2018-05-11 返修日期:2018-07-11 本文受国家自然科学基金:基于粒计算的多源异构动态数据挖掘关键技术研究项目(61573292)资助。

朱立霞(1993—),女,硕士生,CCF 会员,主要研究方向为大数据可视化,E-mail:251856315@qq.com;李天瑞(1969—),男,博士,教授,CCF 会员,主要研究方向为云计算、数据挖掘、人工智能等,E-mail:trli@swjtu.edu.cn(通信作者);滕飞(1984—),女,博士,教授,CCF 会员,主要研究方向为云计算和工业大数据挖掘;彭博(1980—),女,博士,副教授,主要研究方向为计算机视觉和模式识别。

## 2 相关工作

### 2.1 时序数据可视化

人们常把历史和时间比喻成长河,因为人们所能感知的时间是线性的,是有着明确的指向性的。因此,在展示时序数据时,常以两个维度来展示:一个维度表示时间,另一个维度表示数据随时间的变化情况。例如,Weber等<sup>[1]</sup>于2001年设计了一种基于螺旋形时间轴的可视化方法,其通过交互选择周期长度还可以揭示数据的变化模式。若时间影响的数据是多维的,则可以采用Tominski等<sup>[2]</sup>于2004年提出的基于平行坐标的可视化展示方法。如图1所示,他们在位于中心的时间轴的周围放置其他维度轴,以便能够清晰地观察到多个维度的数据的分布情况。近几年,可视化技术在展示时间序列数据方面更是取得了卓越的成果,不仅是传统的平行坐标等可视化方法得到了完善,还出现了曲线图<sup>[3]</sup>、桑基图<sup>[4]</sup>和管状图<sup>[5]</sup>等。其中,展示时间序列数据时,螺旋图的应用尤为广泛。这主要因为通过连续的螺旋线,既能展现数据在时间上的连续性,又能展现数据在不同时间周期上的变化规律。例如,杨欢欢等<sup>[6]</sup>于2017年提出了一种螺旋图可视化方法来展示时间序列数据,该可视化方法使用圆环表示状态,采用一组同心圆环来表示一个事件的多个状态,螺旋线表示相邻状态间的可达路径。螺旋图的可视化效果如图2所示,它不仅能够在有限的空间中有效地展示数据,还解决了时间序列长度对空间消耗的不确定性问题,而且理论上能在有限的空间展示任意时长的路径。

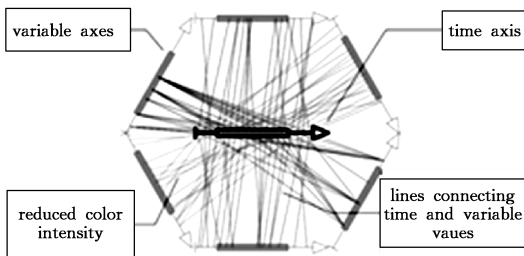


图1 基于平行坐标的可视化方法<sup>[2]</sup>

Fig. 1 Visualization method based on parallel coordinates<sup>[2]</sup>

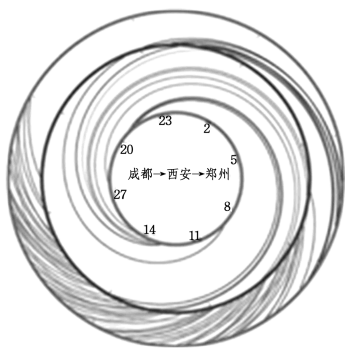


图2 从成都经西安到达郑州的路线数据效果图<sup>[6]</sup>

Fig. 2 Effect map of railway route data from Chengdu to Zhengzhou via Xi'an<sup>[6]</sup>

### 2.2 边绑定的发展

边是可视化技术展示数据的基本元素之一。边的处理方法将影响最终的展示效果,也就是说,边越复杂,可视化效果

也越复杂。边绑定的目的是对相似的边进行分组和变换,以对原始的边缘杂乱的可视化效果提供抽象化和整洁化的新视图<sup>[7]</sup>,在可视化领域中常被用来解决视觉杂乱问题。当数据量达到一定程度时,边与边彼此会离得很近,此时称它们是“相容的”。边绑定就是将这些“相容的”边合并成一个包,让它们彼此共享路径<sup>[8]</sup>。边捆绑的展示结果是一个简化的图,它减少了节点间的多个关系和连接路径,降低了视觉杂乱程度。

应用于平行坐标可视化方法的边捆绑发展迅速。McDonnell等<sup>[9]</sup>于2008年在平行坐标中应用了边捆绑算法,主要是用Bézier曲线代替原平行坐标中的折线,并通过调整控制点的位置来调整曲线的弯曲度,以此调整边捆绑的强度,从而使属于同一分组的曲线相互聚拢,达到节省屏幕空间和缓解视觉杂乱的目的。同年,Zhou等<sup>[10]</sup>另辟蹊径,假设平行坐标的边与边之间有相互作用力,边在其他边的作用力下可弯曲,边的弯曲程度由其所受的其他边的合力决定。该算法不仅最大限度地减小了边的曲率,还最大化了边与相邻边之间的平行度,最终达到边捆绑的效果,有效地缓解了视觉杂乱问题。Heinrich等<sup>[11]</sup>于2011年也使用Bézier曲线代替原有折线,但提出了新的确定控制点的方法,即在2个相邻的属性轴中间假设一条虚拟轴,对边与虚拟轴的交点进行聚类,把聚类中心作为Bézier曲线控制点,这样可以使属于同一个群集的边相互靠近,从而缓解视觉杂乱问题。Palmas等<sup>[12]</sup>于2014年使用高斯核密度估计<sup>[13]</sup>来对平行坐标每个属性轴上的数据点单独聚类,同时在边捆绑之前,在两个相邻的属性轴之间为每个属性轴各添加一条虚拟绑定轴;然后结合原始数据集将相邻2个属性轴上的群集连接,在渲染时使用多边形来代替相邻2个虚拟绑定轴之间的连接关系。这样不仅达到了缓解视觉杂乱的效果,并且大大增加了边的捆绑长度,使整个边绑定效果更加清晰。

## 3 基于区间分类的边绑定算法

本文设计了一种边绑定算法,以解决传统螺旋图在展示大量数据时出现的视觉杂乱问题。本节将介绍本文的边绑定算法的设计方案,该算法对每一个状态的所有数据点进行分区,对每一对相邻状态圆环间的螺旋线进行边绑定操作,最后用两段贝塞尔曲线和一段螺旋线来映射绑定前的螺旋线。

### 3.1 数据集与螺旋图的映射关系

给定数据集  $D = \{d_1, d_2, \dots, d_p\}$ , 其中  $p$  为数据量;  $d_i = (sd_i, sd_{i+1}, lines_i)$  ( $1 \leq i \leq p$ )。  $sd_i$  和  $sd_{i+1}$  分别表示状态  $i$  和状态  $i+1$  的描述性数据,如状态名称;  $lines_i$  表示从状态  $i$  转换到状态  $i+1$  的路径数据;  $lines_i = (l_{i,1}, l_{i,2}, \dots, l_{i,f_i})$ , 其中  $f_i$  表示从状态  $i$  转换到状态  $i+1$  的路径数量。

如图3所示,同心状态圆环的圆心坐标为  $o = \{x_0, y_0\}$ , 状态圆环集为  $S = \{s_1, s_2, \dots, s_q\}$ ,  $q$  ( $q = p + 1$ ) 为状态圆环的数量。状态圆环的一周表示一个时间周期  $T$ 。第  $g$  ( $1 \leq g \leq q$ ) 个状态圆环  $s_g$  的半径为  $r_g$ , 以坐标  $(x_0, y_0 + r_g)$  表示状态圆环  $s_g$  的周期起点,从起点开始按顺时针方向旋转一周即为一个周期  $T$ 。数据  $d_i$  中的  $sd_i$  对应状态圆环  $s_g$ ,  $sd_{i+1}$  对应状态圆环  $s_{g+1}$ , 且  $g = i$ ,  $lines_i$  映射为  $s_g$  和  $s_{g+1}$  之间的一组螺旋

线;螺旋线的起点定位在  $s_g$  上,终点定位在  $s_{g+1}$  上。

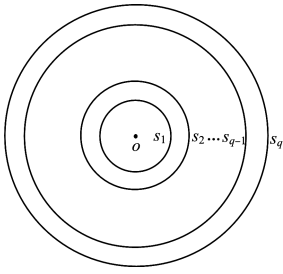


图 3 状态圆环集示意图

Fig. 3 Diagram of state circles

### 3.2 数据分类

目前,大多数的边绑定算法都使用聚类方法来对属性轴上的数据点进行聚类。常用的聚类方法有 Gaussian 核密度、K-means 或 K-means++ 等,如秦红星等<sup>[14]</sup>使用 K-means 聚类算法<sup>[15]</sup>对每个属性轴上的数据点进行聚类。但是,由于本文中状态圆环(类似平行坐标中的属性轴)表示的是一个连续时间区间,人们往往关注的也是某一个连续时间区间内的数据的变化趋势,因此本文不再采取聚类方法来对状态圆环上的原始数据点进行处理,而是直接将状态圆环划分为多个子区间,然后根据数据点所在的子区间来对数据进行分类。具体步骤为:

1) 对于所有的状态圆环,根据划分粒度  $k$ ,从周期起点开始,沿顺时针方向,将状态圆环的时间周期  $T$  划分为  $k$  个相等的子区间,则每个子区间的周期长度为  $T/k$ 。

2) 对于所有的状态圆环,根据状态圆环  $s_g$  上的数据点位于的子区间,将所有数据点分为  $k$  个类别。

图 4 是以  $k=4$  为条件对状态圆环的区间及数据点进行分类的示意图。

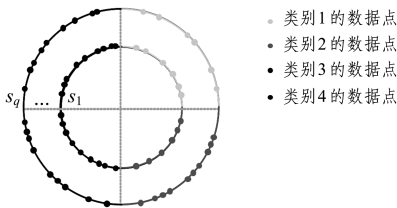


图 4 对数据点进行分类的示意图

Fig. 4 Schematic diagram for classifying data points

### 3.3 创建捆绑关系

在传统的螺旋图中,数据点用螺旋线连接。本文将用 Bézier 曲线与螺旋线结合的方式代替传统的螺旋线。在捆绑每个类别的数据边之前,首先要为其构建捆绑关系。我们设划分粒度  $k=4$ ,以两个相邻的状态圆环  $s_g$  和  $s_{g+1}$  中的一条螺旋线为例来介绍创建绑定关系的实现过程。

首先,在状态圆环  $s_g$  和  $s_{g+1}$  之间放置  $s_g$  对应的虚拟绑定圆环  $s_g^r$ ,以及  $s_{g+1}$  对应的虚拟绑定圆环  $s_{g+1}^l$ ,将状态圆环间的区域划分为 3 个部分,如图 5 所示。虚拟绑定圆环与对应的状态圆环的距离根据距离比  $\beta$  来确定:

$$dis_g = dis_{g+1} = bw_{g,g+1} * \beta \quad (1)$$

其中,  $0 < \beta < 0.5$ ,  $bw_{g,g+1}$  为  $s_g$  与  $s_{g+1}$  的半径差,  $dis_g$  表示  $s_g$

与  $s_g^r$  的距离,  $dis_{g+1}$  表示  $s_{g+1}$  与  $s_{g+1}^l$  的距离。为简化实验,本文只考虑  $\beta$  值为 0.1 的情况。

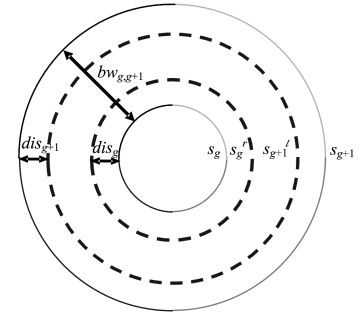


图 5 虚拟绑定圆环示意图

Fig. 5 Schematic diagram of virtual bundling ring

然后,根据绑定强度系数  $\alpha$ ,计算状态圆环  $s_g$  和  $s_{g+1}$  的数据点在其对应的虚拟绑定圆环上的控制点。

考虑到圆环的特殊性,在计算状态圆环的数据点在对应的虚拟绑定圆环上的控制点时,本文采取数据点的偏转角的线性变换方法来实现。如图 6 所示,状态圆环上的某个类别所处的区间用偏转角可表示为三项式  $(\theta_{min}, \theta_{mid}, \theta_{max})$ ,经过线性变换之后,其在对应的虚拟绑定圆环上也表示一个类别,不过是一个更小的区间,用偏转角可表示为三项式:

$$(\theta_{mid} + \alpha(\theta_{min} - \theta_{mid}), \theta_{mid}, \theta_{mid} + \alpha(\theta_{max} - \theta_{mid}))$$

其中,  $\alpha (0 < \alpha < 1)$  是绑定强度系数。

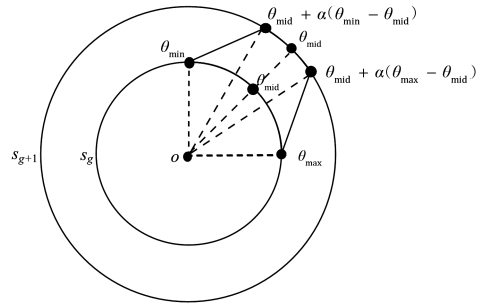


图 6 捆绑角度变化示意图

Fig. 6 Diagram of change of bundling angle

根据以上方法,设螺旋线  $e_{i,j}$  在状态圆环  $s_g (g=i)$  上的位置为  $sT_{kj}$ ,偏转角为  $s\theta_{kj}$ ;在状态圆环  $s_{g+1}$  上的位置为  $aT_{kj}$ ,偏转角为  $a\theta_{kj}$ 。根据式(2)计算出  $s\theta_{kj}$  在虚拟绑定圆环  $s_g^r$  上的偏转角  $s\theta'_{kj}$ ;根据式(3)计算出  $a\theta_{kj}$  在虚拟绑定圆环  $s_{g+1}^l$  上的偏转角  $a\theta'_{kj}$ 。

$$s\theta'_{kj} = s\theta_{g,mid} + \alpha(s\theta_{kj} - s\theta_{g,mid}), 0 < \alpha < 1 \quad (2)$$

$$a\theta'_{kj} = a\theta_{g,mid} + \alpha(a\theta_{kj} - a\theta_{g,mid}), 0 < \alpha < 1 \quad (3)$$

其中,  $s\theta_{g,mid}$  和  $a\theta_{g,mid}$  分别表示数据点  $sT_{kj}$  和数据点  $aT_{kj}$  所在的周期子区间的中点位置的偏转角。

根据式(4)和式(5)计算出数据点  $sT_{kj}$  在  $s_g^r$  上的控制点位置  $sT'_{kj}$ ,根据式(6)和式(7)计算出  $aT_{kj}$  在  $s_{g+1}^l$  上的控制点位置  $aT'_{kj}$ ,如图 7 所示。

$$sT'_{kj}(x) = x_0 + r_g * \cos\left(\frac{2\pi * s\theta'_{kj}}{360}\right) \quad (4)$$

$$sT'_{kj}(y) = y_0 - r_g * \sin\left(\frac{2\pi * s\theta'_{kj}}{360}\right) \quad (5)$$

$$aT'_{gij}(x) = x_0 + r_{g+1} * \cos\left(\frac{2\pi * a\theta'_{gij}}{360}\right) \quad (6)$$

$$aT'_{gij}(y) = y_0 - r_{g+1} * \sin\left(\frac{2\pi * a\theta'_{gij}}{360}\right) \quad (7)$$

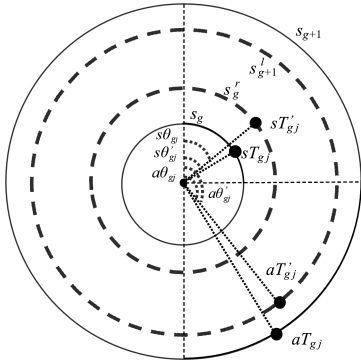


图7 虚拟绑定圆环上的控制点示意图

Fig.7 Diagram of control points on virtual bundling ring

### 3.4 绘制捆绑曲线

对于这一对数据点( $sT'_{gij}$ ,  $aT'_{gij}$ ),传统的螺旋图用螺旋线将它们直接相连,如图8中的曲线 $e_{gij}$ ;然而在本文的边绑定算法下,将用3段曲线来共同展示,如图8所示,第一段是从 $sT'_{gij}$ 到 $sT'_{gij}$ 的Bézier曲线,第二段是从 $sT'_{gij}$ 到 $aT'_{gij}$ 的螺旋线,第三段是从 $aT'_{gij}$ 到 $aT'_{gij}$ 的Bézier曲线,如 $e'_{gij}$ 。

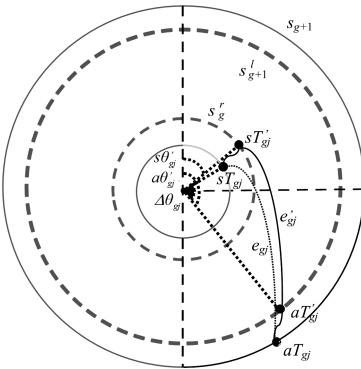


图8 经过边绑定后的效果图

Fig.8 Effect diagram after edge bundling

根据Bézier曲线的绘制公式可以很容易地绘制出两段Bézier曲线,这里不再赘述。为了绘制从 $sT'_{gij}$ 到 $aT'_{gij}$ 的螺旋线,我们需要借鉴文献[16]中的方法,首先计算 $s\theta'_{gij}$ 和 $a\theta'_{gij}$ 的角度差 $\Delta\theta_{gij}$ ,然后求出即将绘制的螺旋线的旋转弧度 $\Delta r_{dian_{gij}}$ ,最后绘制 $sT'_{gij}$ 到 $aT'_{gij}$ 的螺旋线。公式如下:

$$\Delta\theta_{gij} = a\theta'_{gij} - s\theta'_{gij} \quad (8)$$

$$t'_{gij} = \frac{\Delta\theta_{gij}}{360} * T \quad (9)$$

$$\Delta r_{dian_{gij}} = \frac{t'_{gij}}{T} * 2\pi \quad (10)$$

## 4 实验与分析

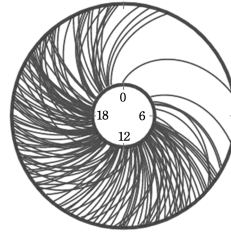
### 4.1 数据集

本文实验选用了中国铁路数据集 stationLines<sup>1)</sup>(包含“成

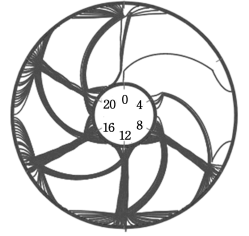
都—西安—郑州”“成都—西安—北京”“北京—石家庄”的可行方案数据)。

### 4.2 验证边绑定算法是否能缓解视觉杂乱

用传统的螺旋图方法展示数据集 stationLines 中“北京—石家庄”的可行方案,得到如图9(a)所示的效果图。用本文的方法进行展示,设置参数 $\alpha=0.5$ , $k=6$ ,效果如图9(b)所示。可以看出,图9(a)中状态圆环 $s_1$ 和 $s_2$ 之间的螺旋线几乎无法直接用肉眼进行分辨,而经过上述参数设置的边绑定之后,我们从图9(b)中可以很容易区分不同时间区间内的螺旋线,并且大大缩小了所有螺旋线占用的屏幕空间,整个界面非常简洁美观。



(a)传统螺旋图可视化方法的效果图



(b)经过边绑定之后的效果图

图9 绑定效果的比较

Fig.9 Comparison of bundling effects

### 4.3 不同划分粒度下边绑定的可视化效果

图10分别在3种情形中展示“北京—石家庄”的可行方案数据:图10(a)展示参数 $\alpha=0.5$ , $k=4$ 的情形;图10(b)展示参数 $\alpha=0.5$ , $k=6$ 的情形;图10(c)展示参数 $\alpha=0.5$ , $k=8$ 的情形。实验表明,对于同一数据、同一绑定强度 $\alpha$ ,不同的划分粒度 $k$ 均有很明显的绑定效果,都能降低状态圆环之间的路径相互交叉或者重叠的现象。

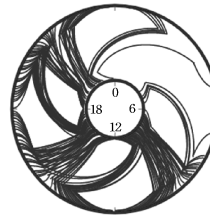
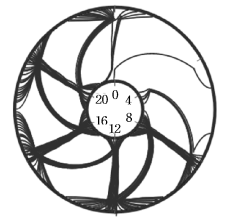
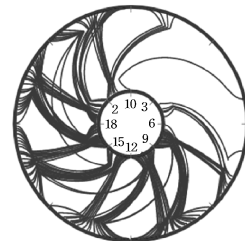
(a)参数 $\alpha=0.5$ , $k=4$ (b)参数 $\alpha=0.5$ , $k=6$ (c)参数 $\alpha=0.5$ , $k=8$ 

图10 不同划分粒度下的绑定的效果图

Fig.10 Bundling effect diagram under different partition granularity

### 4.4 不同数据集下边绑定的可视化效果

在参数 $\alpha=0.5$ , $k=4$ 的情况下,图11分别展示了“成都—西安—郑州”的可行方案数据(见图11(a))、“成都—西

<sup>1)</sup> <http://www.12306.cn/mormhweb/>

安—北京”的可行方案数据(见图 11(b))和“北京—石家庄”的可行方案数据(见图 11(c))。实验表明,对于同一绑定强度  $\alpha$  和同一划分粒度,不同的数据集均有较好的绑定效果,都能降低状态圆环之间的路径相互交叉或者重叠的现象。

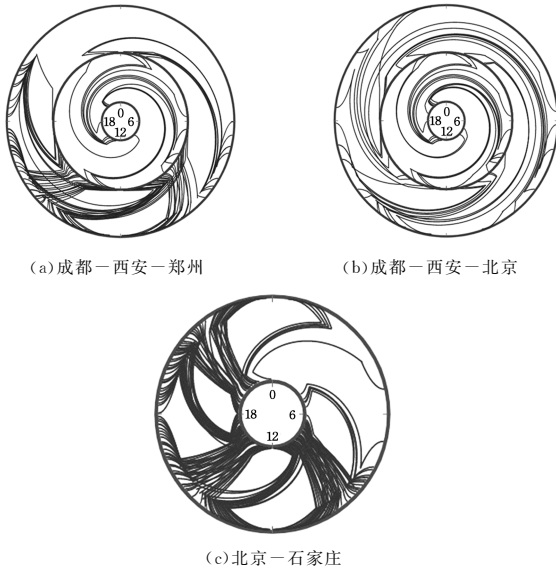


图 11 不同数据集下的绑定效果图

Fig. 11 Bundling effect diagram under different data sets

**结束语** 针对现有螺旋图可视化方法在展示大量数据时会出现因螺旋线交叉而导致视觉杂乱的问题,本文提出了一种将数据分类,然后根据分类结果将同一类螺旋线绘制成束,以实现边绑定效果的螺旋图边绑定算法,从而提高了螺旋图中螺旋线的视觉聚集程度,缓解了视觉杂乱问题,进一步提高了可视化效果。

在后续的研究中,还可以在以下几个方面进一步优化本文所提出的工作:1)在我们目前的实验中,尚未充分研究所有可用于边绑定算法的曲线类型,无法得知是否有比本文中的曲线更合适的方法;2)在本文的实验中,交互主要是通过用户点击系统提供的选项来调整整个可视化展示的内容,在接下来的工作中,可以考虑使用更复杂的交互工具(如刷技术)来进一步扩展系统。

## 参 考 文 献

[1] WEBER M, ALEXA M, MÜLLER W. Visualizing Time-Series on Spirals[C]//Proceedings of IEEE Symposium on Information Visualization. IEEE, 2001: 7-13.

[2] TOMINSKI C, ABELLO J, SCHUMANN H. Axes-based visualizations with radial layouts[C]//Proceedings of ACM Symposium on Applied Computing. ACM, 2004: 1242-1247.

[3] LEI H, XIA J, GUO F, et al. Visual exploration of latent ranking evolutions in time series[J]. Journal of Visualization, 2016, 19

(4): 1-13.

[4] JIANG T T, XIAO W D, ZHANG C, et al. Text visualization method for time series based on Sankey diagram[J]. Application Research of Computers, 2016, 33(9): 2683-2687. (in Chinese) 姜婷婷, 肖卫东, 张翀, 等. 基于桑基图的时间序列文本可视化方法[J]. 计算机应用研究, 2016, 33(9): 2683-2687.

[5] BOUALI F, DEVAUX S, BASTIEN, et al. Visual mining of time series using a tubular visualization[J]. Visual Computer, 2016, 32(1): 15-30.

[6] YANG H H, LI T R, CHEN X D. Visualization of time series data based on spiral graph[J]. Journal of Computer Applications, 2017, 37(9): 2443-2448. (in Chinese) 杨欢欢, 李天瑞, 陈馨葭. 基于螺旋图的时间序列数据可视化[J]. 计算机应用, 2017, 37(9): 2443-2448.

[7] ZHOU H, XU P, YUAN X, et al. Edge Bundling in Information Visualization[J]. Tsinghua Science and Technology, 2013, 18(2): 145-156.

[8] GANSNER E, HU Y, NORTH S, et al. Multilevel agglomerative edge bundling for visualizing large graphs[C]//Proceedings of IEEE Pacific Visualization Symposium. IEEE, 2011: 187-194.

[9] MCDONNELL K T, MUELLER K. Illustrative parallel coordinates[J]. Computer Graphics Form, 2008, 27(3): 1031-1038.

[10] ZHOU H, YUAN X R, QU H M, et al. Visual clustering in parallel coordinates[J]. Computer Graphics Forum, 2008, 27(3): 1047-1054.

[11] HEINRICH J, LUO Y, KIRKPATRICK A E, et al. Evaluation of a bundling technique for parallel coordinates[J]. Energy Conversion & Management, 2011, 88(5): 259-266.

[12] PALMAS G, BACHYNSKYI M, OULASVIRTA A, et al. An edge-bundling layout for interactive parallel coordinates[C]//Proceedings of IEEE Pacific Visualization Symposium. IEEE, 2014: 57-64.

[13] SPURR B D. Density estimation for statistics and data analysis [J]. Journal of the Royal Statistical Society, 1987, 150(4): 403-404.

[14] QIN H X, WEI X S. A Study on Edge Bundling Technology in Parallel Coordinates[J]. Journal of Computer-Aided Design & Computer Graphics, 2017, 29(7): 1235-1244. (in Chinese) 秦红星, 卫学仕. 平行坐标中的边捆绑算法[J]. 计算机辅助设计与图形学学报, 2017, 29(7): 1235-1244.

[15] KANUNGO T, MOUNT D M, NETANYAHU N S, et al. An efficient k-means clustering algorithm: analysis and implementation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(7): 881-892.

[16] HOLTEN D, VAN WIJK J J. Evaluation of Cluster Identification Performance for Different PCP Variants[J]. Computer Graphics Forum, 2010, 29(3): 793-802.