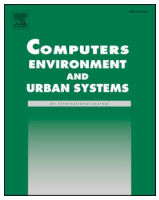
Computers, Environment and Urban Systems 122 (2025) 102338

Contents lists available at ScienceDirect

Computers, Environment and Urban Systems

journal homepage: www.elsevier.com/locate/ceus



ScaleFC: A scale-aware geographical flow clustering algorithm for heterogeneous origin-destination data

Huan Chen [a], Zhipeng Gui [b,c,d,e,*], Dehua Peng [b,d,e,*], Yuhang Liu [a], Yuncheng Ma [a], Huayi Wu [a,d,e]

[a] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China
[b] School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China
[c] School of Geography and Planning, Ningxia University, Yinchuan, China
[d] Collaborative Innovation Center of Geospatial Technology, Wuhan University, Wuhan, China
[e] Hubei Luojia Laboratory, Wuhan, China

ARTICLE INFO

Keywords:
Geographical flow
Flow clustering
Flow pattern analysis
Heterogeneous distribution
Spatial scale

ABSTRACT

Exploring the cluster pattern of geographical flow facilitates the understanding of the underlying process of geographical phenomena among spatial locations. Despite recent advancements in identifying flow clusters, challenges remain when handling flow data with uneven length, heterogeneous density and weak connectivity. To solve the issues, this study proposes a Scale-aware Flow Clustering algorithm (ScaleFC). It identifies flow clusters of arbitrary lengths by employing an analytical scale to generate an adjustable neighborhood range of each flow. Meanwhile, inspired by the idea of boundary-seeking clustering, ScaleFC introduces partitioning flows to identify flow clusters with different densities, and separate the weakly-connected clusters. To validate the effectiveness, we compared ScaleFC with three mainstream baselines, i.e., AFC, FlowLF and FlowDBSCAN, on six synthetic datasets. The results presented that ScaleFC can accurately identify the clusters with complex structures, achieving an average accuracy improvement of 27 %, 17 %, and 15 % over the three competitors, respectively. The application on bike-sharing data with 16,140 flow pairs from Shanghai City demonstrated that ScaleFC is capable to capture both long-distance and short-distance movements, thereby providing a more comprehensive understanding to multi-scale human mobility patterns in geographical space.

1. Introduction

Geographical flow, also known as origin-destination (OD) flow, is a crucial type of geospatial data that describes the movements, interactions, and connections between different geospatial locations (Pei et al., 2020). Identifying clusters in geographical flow contributes to reflecting the frequent mobility patterns of geographical objects, and understanding the underlying mechanisms of spatial interactions. For example, the taxi flow clusters during rush hour uncover active traffic routes and reveal home-work commuting patterns, thus providing support for transportation planning decisions (Yan et al., 2022). Similarly, analyzing clusters in bike-sharing flows highlights the travel demands of crowds and hotspots in residential areas. It offers valuable insights for optimizing the allocation and distribution of shared bicycles to better meet the short-distance travel needs of urban residents (He et al., 2018). Therefore, detecting flow clusters has great potential in urban management (Andrienko et al., 2017; Yan et al., 2021), transportation planning (Liu, Gui, et al., 2024; Liu, Li, et al., 2024; Nielsen & Hovgesen, 2008), individual mobility prediction (Li et al., 2020; Wesolowski et al., 2012) and etc.

Despite recent advancements in flow clustering, existing approaches are mainly extended from traditional spatial point clustering techniques, and face challenges in identifying complex flow patterns (Guo et al., 2025). The spatial distribution of natural flows exhibits inherent complexity characterized by uneven length (Tao & Thill, 2016b), heterogeneous density (Liu, Yang, Deng, Song, & Liu, 2022), and weak connectivity (Tang et al., 2024). These characteristics present challenges for both flow cluster identification and the design of flow clustering algorithms. Specifically, ***uneven length*** refers to the presence of flows with substantial differences in length, indicating the movements over varying distances. This phenomenon makes it difficult to simultaneously capture both long- and short-distance flow clusters using a fixed

* Corresponding authors at: School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China.
E-mail addresses: zhipeng.gui@whu.edu.cn (Z. Gui), pengdh@whu.edu.cn (D. Peng).

https://doi.org/10.1016/j.compenvurbsys.2025.102338
Received 11 November 2024; Received in revised form 26 July 2025; Accepted 10 August 2025

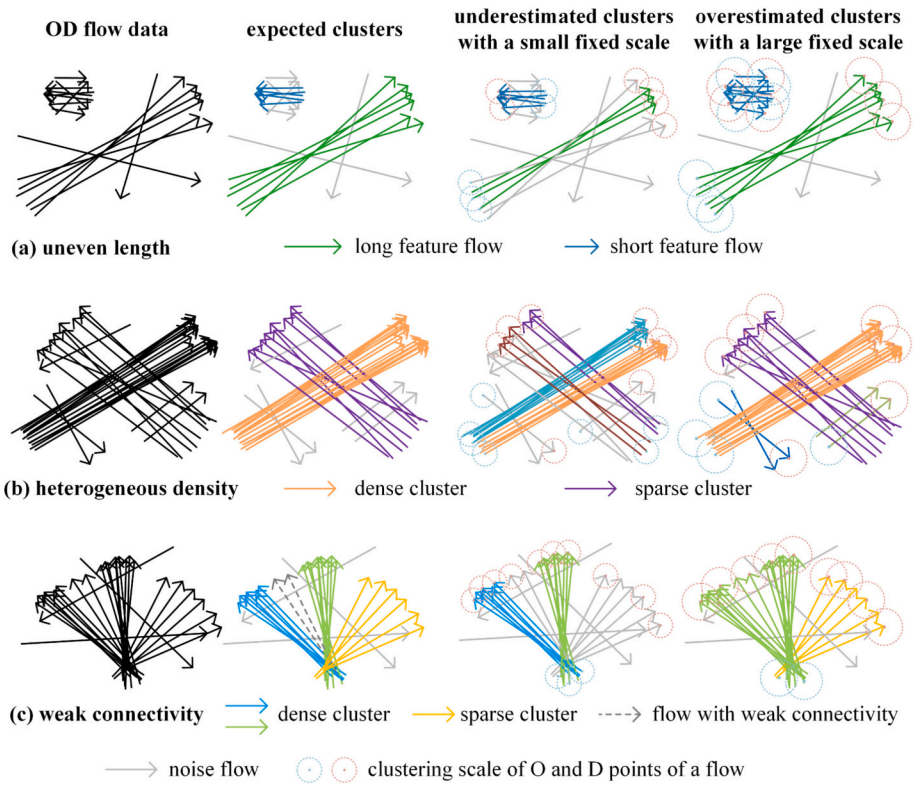


Fig. 1. Flow clustering on three toy datasets that contain flows with (a) uneven length, (b) heterogeneous density and (c) weak connectivity, respectively. The legend is to be read with the expected clusters.

clustering scale. As illustrated in Fig. 1 (a), a fixed small or large scale might misidentify the long feature flows (Song et al., 2020) as noise, or group overlapped short flows with distinct directions into a single cluster, leading to underestimated or overestimated clustering results, respectively. ***Heterogeneous density*** in OD flows refers to the coexistence of both dense and sparse OD flows across space caused by the discrepancy in mobility frequency. It poses challenges for density-based clustering methods on density threshold parameter settings to accurately identify both dense and sparse clusters. As shown in Fig. 1 (b), a low density threshold may obtain underestimated clusters, leading to the sparse clusters being misidentified as noise and producing many tiny subclusters; while a high threshold causes the noise flows to be incorrectly grouped into valid clusters. ***Weak connectivity*** refers to the phenomenon that nearby high-density clusters are connected by sparse or noise flows between them. In Fig. 1 (c), for connectivity-based clustering methods, two dense flow clusters in blue and green color are likely to be merged together when using an insufficient small clustering scale (Tran et al., 2013). However, a small scale may cause the sparse clusters in orange color to be ignored. Overall, the above clustering issues hinder from revealing the real movements that flow clusters present, resulting in invalid mobility pattern mining and interpretation.

To address the aforementioned issues in heterogeneous OD data, we propose a scale-aware flow clustering algorithm (ScaleFC). The main contributions of this work are as follows:

- A scale factor is developed to adaptively specify the neighborhood range of each flow based on its length when searching neighbors. Such a scale-aware mechanism can address the uneven length issue to achieve the extraction of flow cluster patterns at both macro and micro scales.
- Inspired by boundary-seeking clustering, we introduce partitioning flows as boundaries between clusters by perceiving local density variations among individual flows. This approach facilitates separating nearby clusters to cope with weak connectivity in flows.
- Combining the above two mechanisms can identify clusters with heterogeneous densities. The scale factor mitigates sparse long-distance flow clusters to be misclassified as noise, while partitioning flows are able to split adjacent clusters with varying densities.

We design six synthetic datasets characterizing uneven length, heterogeneous density and weak connectivity in this study. Additionally, we conduct extensive comparative experiments and parameter analysis using these synthetic datasets, and empirical analysis on a real-world bike-sharing OD dataset from Shanghai City. The source code for ScaleFC and the three comparison flow clustering algorithms, along with these datasets, are publicly shared to support further research (https://github.com/ZPGuiGroupWhu/ScaleFC).

The remainder of this paper is organized as follows. Section 2 reviews the related studies. Section 3 provides a detailed algorithm description. Section 4 presents validation on synthetic datasets. Section 5 demonstrates a case study on a bike-sharing OD dataset. Section 6 summarizes the study and suggests future directions.

2. Related work

The study on geographical flow clustering can be categorized into aggregated and individual flow clustering based on the type of flow data used (Tao et al., 2023). The former partitions the study area into predefined geographical units, i.e., grids, hexagons or demarcated units, and counts OD flow pairs between units to identify hotspots using a threshold. However, these approaches suffer from the modifiable areal unit problem (MAUP), and ignore the short flows within the units (Zhu & Guo, 2014). The latter utilizes precise geographical coordinates of O and D points to cluster flows based on spatial similarity measurements. Dispensing with the construction of geographical units, these methods can capture more complex and fine-grained flow patterns. Therefore, we focus on individual flow clustering in this study. Existing individual geographical flow clustering can be further divided into three

categories, i.e., hierarchical, statistics-based, and density-based methods (Pei et al., 2020).

2.1. Hierarchical flow clustering

Similar to traditional hierarchical clustering for point data, both agglomerative and divisive approaches are utilized for flow clustering. These methods measure similarity between individual flows based on the proximity of flow attributes, e.g., endpoint distance, length, direction, density and etc. Then, they employ a bottom-up linkage strategy to merge similar flows into a cluster at different scales step-by-step, or use a top-down divisive strategy to divide the whole flow data into different flow sub-clusters. For agglomerative clustering, flow similarity is calculated based on shared-nearest-neighbor (SNN) density of O and D points, and flows with high similarity are aggregated (Zhu & Guo, 2014). Such SNN-based measurement improves robustness in identifying flow clusters with varying densities. Although flows can be generalized to different hierarchical levels, the SNN similarity ignores the shape of flow distribution, resulting in inaccurate clustering outcomes and loss of true flow patterns (Liu, Yang, Deng, Song, & Liu, 2022). To solve this problem, geometric attributes of flows, such as length and direction, are adopted to calculate inter-flow similarity (Yao et al., 2018). Alternatively, flows are treated as vectors, with similarity defined based on multi-dimensional constraints in vector space. It incorporates features like frequency and intensity to capture a broader range of flow characteristics (Guo et al., 2020). These flow clustering methods that consider flow length in similarity measurements are helpful to tackle the uneven flow length issue. For example, a similarity metric is defined as the sum of squares of the O-pair and D-pair distances, divided by the product of flow lengths (Tao & Thill, 2016b, 2019a). However, it cannot intuitively capture the travel distance of spatial objects in geographical space and doesn't satisfy the triangle inequality, thereby limiting its applicability for spatial statistics of flows (Pei et al., 2020).

As for the divisive flow clustering, the grid (Wood et al., 2009), Delaunay triangulation (Adrienko & Adrienko, 2011) and Voronoi diagram (Guo et al., 2012) are commonly used to partition the spatial domain containing flow data. Then, high-density cells are aggregated and recognized as clusters. Another widely-used approach in divisive flow clustering involves graph-based pruning to identify flow clusters. For example, greedy pruning strategy is used to divide minimum spanning tree (MST) constructed from flow distance matrix by iteratively removing edges between nodes with the greatest distances (Tao et al., 2017). Flow clusters are then determined based on a predefined threshold for the minimum number of flows. Maximum spanning tree (MaxST), is also employed to implement a recursive two-way optimum approach to partition the flow data into clusters, with a child graph self-similarity criterion to determine the clustering results (Xiang & Wu, 2019). These graph-based flow clustering algorithms leverage global structural connectivity and local density-aware mechanisms to tackle the issue of heterogeneous density. However, their adoption of a fixed clustering scale is inadequate for solving uneven length and weak connectivity issues.

In summary, hierarchical flow clustering can obtain flow cluster results at different scales to reveal the nested spatial structures of flow data. Specifically, agglomerative methods demonstrate improvement in handling uneven length, while graph-based divisive methods are capable of tackling heterogeneous density. Nevertheless, most hierarchical clustering approaches show limitations in weak connectivity issue. Besides, these methods encounter difficulties in distinguishing between feature flows and noise flows. Noise flows with low densities and irregular spatial distributions distort the data structure, leading to inaccurate clustering and obscuring meaningful patterns (Zhong & Duckham, 2016). Consequently, increased attention has been given to statistics-based and density-based flow clustering methods due to their ability in eliminating noise flows.

2.2. Statistics-based flow clustering

Statistics-based flow clustering extends spatial statistical indicators (Anselin, 1995; Ripley, 1976) from point data to flow data to measure the spatial association and heterogeneity of flows. By calculating the spatial indicator for each flow and conducting statistical significance test, these methods classify flows into feature flows and noise flows, and then identify hotspot regions with dense flows to form clusters (Cai & Kwan, 2022; Zhou et al., 2023). For instance, the Getis–Ord G statistic is employed with a customized spatial weight matrix to identify hotspot areas and detect flow clusters (Berglund & Karlstrm, 1999). Besides, flows are considered as vectors and the modifiable Moran's I statistics are utilized to quantify the spatial autocorrelation of vectors at global and local scales respectively (Liu et al., 2015). The global indicator evaluates the presence of spatial aggregation in flows across the entire study area; while the local indicator identifies flows with high spatial correlation, which are then grouped into flow clusters. These approaches depend on static autocorrelation measures with a fixed neighborhood range, which restricts their ability to conduct multi-scale clustering analysis (Gui et al., 2020). Recently, spatial statistics methods for analyzing point patterns have been expanded to flow cluster pattern analysis (Shu et al., 2021). These approaches conceptualize flows as points within a four-dimensional Cartesian space formed by origin and destination spaces. Based on complete spatial randomness (CSR), spatial density of flows is measured using second-order analysis (Ripley, 1976). Building on Ripley's K-function for points, flow K-function is proposed (Tao & Thill, 2016b). It develops a flow similarity based on flow lengths to compute local flow K-function for each flow, and employs Monte Carlo simulation to preserve flows with high statistical confidence as flow clusters. Owing to its scale-aware flow distance, this method demonstrates effectiveness in dealing with the uneven length issue. However, this approach fails to quantify the aggregation scales of flows (Shu et al., 2021). To solve the problem, flow L-function is defined to determine the observation radius of flows according to the derivative of global L-function, and then merge flows with high density at the corresponding scale (Shu et al., 2021). This approach shows proficiency in handling heterogeneous density issue by extracting dominant cluster at different scale step by step. Although these methods can effectively dispose noise flows, they rely on rigorous prior statistical assumptions of the flow data distribution. However, the real-world data distributions are always too complex to satisfy these assumptions that makes these methods less effective or infeasible.

Another type of statistics-based flow clustering method uses a moving and resizable spatial scanning window to count the frequency of flows in the window and identifies the flows with high probabilities as clusters. A multi-dimensional Bernoulli spatial scan statistics can detect highly associated OD region pairs (Gao et al., 2018). However, this method is more suitable for flows with regular shape due to using a predefine scanning window (Tang et al., 2024). To identify flow clusters of arbitrary shapes, a multi-directional optimal ecological community-based algorithm (AMOEBA) is extended to detect the hotspot and coldspot areas in large flow data (Tao & Thill, 2019b). Inspired by this method, a novel bivariate flow clustering method named BiFlowAMOEBA is also proposed (Liu, Yang, Deng, Liu, & Xu, 2022). In addition, arbitrarily shaped flow clusters can be detected using a spatial scan statistical approach based on ant colony optimization (Song et al., 2019). These scanning methods can detect flow clusters with statistical significance. However, the clustering results are sensitive to the shape and size of the scanning window, which are often difficult to determine accurately. The geometry of the scanning window also impacts the ability of these methods in handling uneven length, heterogeneous density and weak connectivity issues in flow clustering.

Overall, statistics-based flow clustering methods can find statistically significant flow clusters from the noisy background, and show potential in dealing with heterogeneous flow density by using spatial statistical metrics, local parameter optimization and probability model fitting (Yan

Table 1
Evaluation of flow clustering algorithms in addressing uneven length, heterogeneous density, and weak connectivity issues.

Category	Subcategory	Representative Method	Flow clustering issues			Reference
			Uneven length	Heterogeneous density	Weak connectivity	
Hierarchical	Agglomerative	AFC	☆	☆☆	☆	(Zhu & Guo, 2014)
		/	☆☆	☆☆	☆	(Yao et al., 2018)
	Divisive	TOCOFC	☆	☆☆	☆☆	(Xiang & Wu, 2019)
		/	☆☆	☆	☆	(Guo et al., 2020)
Statistics-based	Spatial autocorrelation	/	☆	☆☆	☆	(Berglund & Karlstrm, 1999)
		/	☆	☆☆☆	☆	(Liu et al., 2015)
	Ripley's statistics	FlowKF	☆☆	☆☆	☆	(Tao & Thill, 2016b)
		FlowLF	☆	☆☆☆	☆	(Shu et al., 2021)
	Spatial scan statistics	/	☆	☆☆	☆	(Gao et al., 2018)
		flowAMOEBA	☆☆	☆☆	☆	(Tao & Thill, 2019b)
		AntScan_flow	☆	☆☆☆	☆	(Song et al., 2019)
Density-based	Density threshold	FlowDBSCAN	☆	☆	☆	(Tao & Thill, 2016a)
	Density adaptation	AF-OPTICS	☆	☆☆	☆	(Guo et al., 2025)
	Density decomposition	/	☆	☆☆☆	☆	(Song et al., 2020)
Hybrid	Hierarchical and density-based	FlowHDBSCAN	☆☆	☆☆	☆	(Tao et al., 2017)
	Statistics- and density-based	SNN_flow	☆	☆☆☆	☆	(Liu, Yang, Deng, Song, & Liu, 2022)
		SDBC	☆	☆☆☆	☆☆	(Tang et al., 2024)

Note: The '/' in the *representative method* field indicates that there is no official algorithm name given in the paper. More stars signify a better capability in addressing the corresponding issues.

et al., 2022). However, they have limits in pinpointing the precise locations of flow clusters and addressing weak connectivity issue, as flow similarity is mainly measured in the statistical space and distance-based connectivity mechanisms are further employed to generate flow clusters. Additionally, these methods are time-consuming due to intensive statistical simulations to meet the statistical confidence requirements. To identify the locations of flow clusters in a precise and efficient manner, density-based flow clustering methods have been developed in recent years.

2.3. Density-based flow clustering

Most density-based methods calculate flow density by identifying spatial neighbors, and use density connectivity mechanisms to group high-density flows (Pei et al., 2015). Unlike statistics-based flow clustering, the clustering results have explicit locations. Meanwhile, the time efficiency of these methods relies less on the extensive iterations of simulations, and they can leverage spatial indices like R-trees to further improve performance. For examples, a multi-scale flow clustering algorithm FlowHDBSCAN is developed upon classical DBSCAN algorithm (Tao et al., 2017); an adaptive OPTICS flow clustering algorithm is also proposed to gain flow clusters at different scales (Guo et al., 2025). Nevertheless, selecting an appropriate clustering scale is challenging due to the differences in cluster shapes and densities, parameter sensitivity to noise and outliers, and the absence of prior knowledge about the data distribution (Liu, Yang, Deng, Song, & Liu, 2022). Therefore, density domain decomposition is utilized to determine the clustering scales (Song et al., 2020). However, this algorithm sacrifices the computational efficiency in the parameter estimation process, as it requires traversing all possible parameters to select the optimum to decompose the flow set. Besides, statistics- and density-based flow clustering methods are also combined to solve the clustering scale selection issue. A SNN-based algorithm is proposed to detect flow clusters for network-constrained OD flows, where the density threshold is modeled as a significance level of a statistical test (Liu, Yang, Deng, Song, & Liu, 2022). A statistical and density-based clustering (SDBC) finds high-density flows using the Getis–Ord G statistic, and performs permutation tests to eliminate the candidate clusters with low statistical significance (Tang et al., 2024).

Although density-based flow clustering has distinct advantages in removing noise flows and identifying arbitrary shaped clusters with precise locations, they are still insufficient to deal with uneven flow length, heterogeneous flow density and weak connectivity. These approaches often adopt a fixed analytical scale to identify neighbors for all flows. They tend to misclassify the long feature flows as noise, or mixed the overlapped short flows with different directions together. In addition, the sparse clusters are easily to be detected as noise, and the weakly-connected clusters cannot be separated due to the smooth transition in density between them.

Table 1 provides an overview evaluation of the aforementioned flow clustering algorithms concerning uneven length, heterogeneous density, and weak connectivity issues. In summary, the single-scale flow clustering methods are difficult to detect long- and short-distance flow clusters simultaneously. While, existing multi-scale clustering methods still suffer from heterogeneous density and weak connectivity. Therefore, this study introduces a novel flow clustering algorithm named ScaleFC to solve these issues in a unified framework. Specifically, we propose a scale-aware neighbor search through modifying the neighborhood range of each flow to tackle uneven length and heterogeneous density issues cause by a fix clustering scale. Besides, we introduce the concept of partitioning flows to seek the boundaries between flow clusters, and use these boundaries to separate clusters with varying densities and weak connectivity.

3. Methodology

This section outlines the methodology of our algorithm ScaleFC. Specifically, we introduce necessary concepts and definitions, and then provide the detailed descriptions of the algorithm. Besides, we analyze the time complexity to evaluate its computational efficiency and provide adaptive methods to estimate the two parameters of ScaleFC. The pseudocode of the algorithm can be found in Appendix A.

3.1. Basic concepts

The following definitions about geographical flow, flow neighbor, and flow group are necessary to understand the subsequent clustering steps in ScaleFC.

Definition 1 (Flow): A geographical flow f with length l and direction θ are denoted as follows:

$$f = < (x^O, y^O), (x^D, y^D) > \quad (1)$$

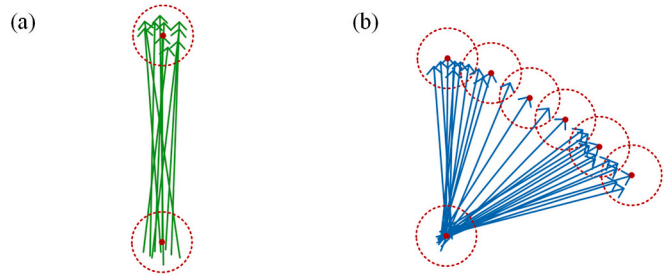


Fig. 2. Illustration of (a) a strongly-connected flow group and (b) a weakly-connected flow group, respectively.

$$l = \sqrt{(x^D - x^O)^2 + (y^D - y^O)^2} \tag{2}$$

$$\theta = \begin{cases} \arctan\dfrac{y^D - y^O}{x^D - x^O} & \text{if } x^D > x^O \text{ and } y^D \geq y^O \\ \arctan\dfrac{y^D - y^O}{x^D - x^O} + 2\pi & \text{if } x^D > x^O \text{ and } y^D < y^O \\ \arctan\dfrac{y^D - y^O}{x^D - x^O} + \pi & \text{if } x^D < x^O \\ \dfrac{\pi}{2} & \text{if } x^D = x^O \text{ and } y^D > y^O \\ \dfrac{3\pi}{2} & \text{if } x^D = x^O \text{ and } y^D < y^O \end{cases} \tag{3}$$

where (x^O, y^O) and (x^D, y^D) are the coordinates of O and D point, respectively. θ is defined in the range $[0, 2\pi)$ with respect to the angle formed with the positive X-axis (Pei et al., 2020).

Definition 2 (Flow maximum distance): Given two flows f_i and f_j, flow maximum distance is the larger one between two O points and that of two D points in the Euclidean distance, defined as follows:

$$D\left(f_i, f_j\right) = \max\left(d_{ij}^O, d_{ij}^D\right) \tag{4}$$

where d_{ij}^O and d_{ij}^D denote the O-pair and D-pair Euclidean distances, respectively. The flow maximum distance is commonly used for measuring flow proximity (Shu et al., 2021).

Definition 3 (Flow neighbor): Given two flows f_i and f_j, f_j is a neighbor of f_i when their spatial distance satisfies that:

$$D\left(f_i, f_j\right) \leq \epsilon \tag{5}$$

where ϵ denotes the distance threshold of the neighborhood around f_i. Based on flow maximum distance, when flow f_j is a flow neighbor of flow f_i, both the O and D points of f_j must fall within the ϵ-neighborhoods of the O and D points of f_i.

Definition 4 (Scale-aware flow neighborhood range) When the dataset contains flows with significantly uneven lengths, using a fixed neighborhood range for all flows is inappropriate. Therefore, a scale factor is developed to dynamically adjust the neighborhood range of each flow for searching flow neighbors. It is designed based on distance-frequency law (Schläpfer et al., 2021): (1) Flow neighborhood range is positively correlated with flow length. (2) The range no longer increases when the flow length is large enough. Given a flow f_i with length l_i, the neighborhood range ϵ_i of f_i is defined as follows:

$$\epsilon_i = \begin{cases} 0.5 \cdot \alpha \cdot MaxLen & \text{if } l_i \geq MaxLen \\ 0.5 \cdot \alpha \cdot l_i & \text{else} \end{cases} \tag{6}$$

where α denotes the scale factor. The range of α is $[0, 1]$ to ensure that the neighborhood buffers around the O and D points do not intersect with each other, as a fundamental requirement in flow clustering (Gao et al., 2020). *MaxLen* denotes the maximum flow length cutoff, which means that when the flow length exceeds *MaxLen*, its neighborhood range no longer increases.

Based on the Definition 4, the neighboring relationships under a fixed range are symmetric, while those based on dynamic range are asymmetric. This asymmetric presents more accurate spatial dependencies between flows with uneven lengths and offers greater flexibility in capturing the analytical scales of flows (Nielsen & Hovgesen, 2008). To be noted that this asymmetry only reflects the neighborhood relationship. Unlike modified flow distance metrics (Tao & Thill, 2016b, 2019a; Xiang & Wu, 2019; Yao et al., 2018), such a scale factor does not affect spatial distribution of the flow data, as the distance metric remains unchanged using the flow maximum distance.

Definition 5 (Flow group): Flow set $F = \{f_1, f_2, \ldots, f_n\}$ is a flow group when each flow $f_i \in F$ has a subgroup $SF_i(f_i \notin SF_i) \subseteq F$ that satisfies the following conditions:

(1) for any flow $f_j \in SF_i$, f_j is a flow neighbor of f_i
(2) $|SF_i| \geq MinFlows$

where *MinFlows* (*mf*) specifies the minimum number of flows required to form a flow group, and $|SF_i|$ is the cardinal number of the subgroup.

Definition 6 (Spatial compactness indicator of flow group): Given a flow group $F = \{f_1, f_2, \ldots, f_n\}$ ($n \geq MinFlows$), its spatial compactness indicator I_F is the Root Mean Square (RMS) of flow maximum distances between each flow in the group and the centroid flow of the group, denoted as follows:

$$I_F = \sqrt{\frac{\sum_{i=1}^{n} D(f_i, \bar{f})^2}{n}} \tag{7}$$

where $\bar{f}$ is the theoretical centroid flow generated by connecting the calculated centroid point of all O points and of all D points. The indicator quantifies the average distance between individual flows and the centroid flow in a group. Since the neighborhood range of the centroid flow indicates the expected maximum distance if the group is strongly-connected, comparing them can detect weak connectivity of a group. Besides, the indicator also aids in identifying partitioning flows by calculating the local compactness (i.e., local density) of the *k*-nearest neighbors of a flow, detailed in Section 3.2.4. The applicability and limits of the indicator are further examined in Appendix C.

Definition 7 (Strongly- and weakly-connected flow groups): Given a flow group $F = \{f_1, f_2, \ldots, f_n\}$ ($n \geq MinFlows$) with heterogeneous density, if $I_F \leq \bar{\epsilon}$, F can be considered as a strongly-connected flow group, otherwise, F is a weakly-connected flow group, where $\bar{\epsilon}$ is the neighborhood range of the centroid flow of the flow group. As shown in Fig. 2, for strongly-connected flow group, the central flow is closely surrounded by its flow neighbors; while the latter includes a branch of dispersed flows radiating outward.

3.2. Workflow of ScaleFC algorithm

The workflow of ScaleFC includes four steps as shown in Fig. 3 (a): (1) Eliminate noise flows and identify flow groups via spatial connectivity measurement; (2) Recognize the strongly-connected flow groups using the spatial compactness indicator defined in Eq. (7). The strongly-connected flow groups are assigned as the final clusters, while the remain groups are treated as weakly-connected flow groups; (3) Identify partitioning flows (PFs) within the generated weakly-connected flow groups to detect potential strongly-connected groups; (4) Reallocate all partitioning flows to nearest flow clusters and output cluster results. Fig. 3 (b) illustrates the algorithm procedure via an example.

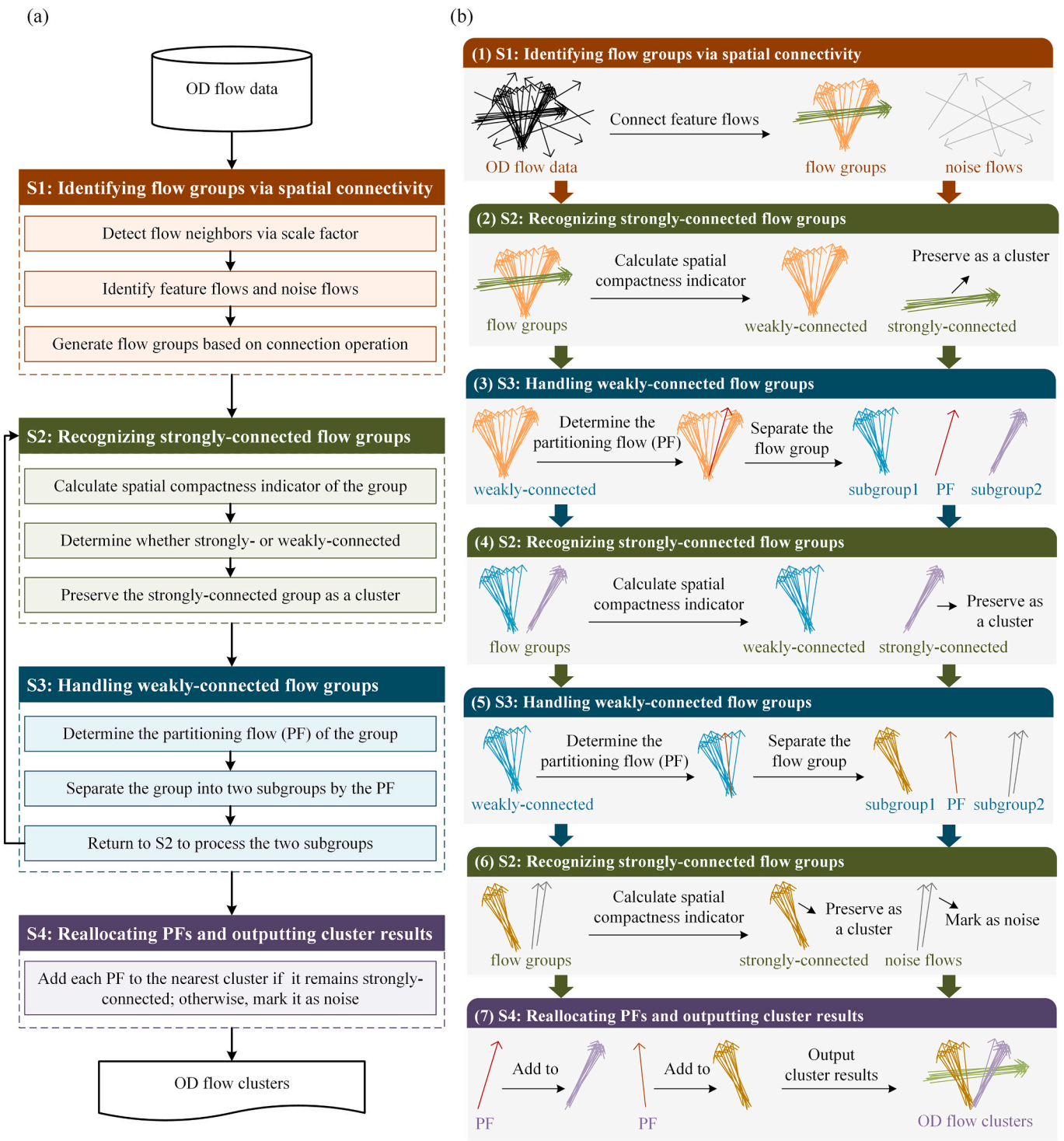


Fig. 3. The illustration of (a) flowchart of ScaleFC and (b) its procedure using an exemplary flow data.

3.2.1. *Identifying flow groups via spatial connectivity*

In this step, we count the number of spatial neighbors of each flow within a specified range and remove the flows whose neighbors are not greater than *MinFlows*. Specifically, the algorithm employs a dynamic neighborhood for each flow to determine local clustering scale adaptively via scale factor (Eq. 6). It groups flows with similar lengths to reduce the influence of length variations on flow clustering results. It also prevents flows with significant length or direction differences from being clustered together due to the transitive nature of the connectivity mechanism. The retained flows are categorized into flow groups through a connection operation. In this operation, each flow is initially treated as an independent group. Then, an unprocessed flow is randomly selected, and merged with its neighbors into a same group. This process continues until all retained flows are processed. As shown in Fig. 3 (b) (1), during this initial processing phase, potential feature flows are aggregated into two distinct flow groups, and the noise flows are removed.

3.2.2. *Recognizing strongly-connected flow groups*

For each obtained flow group, if the number of flows within it is not greater than *MinFlows*, it is also marked as noise flows. For example, in

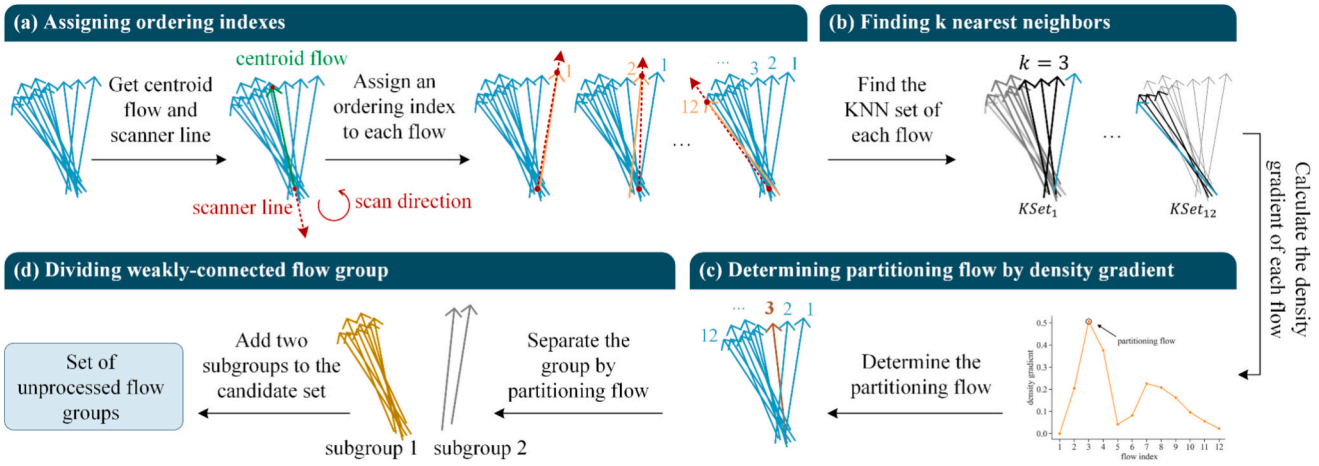


Fig. 4. Workflow for separating a weakly-connected flow group by the partitioning flow.

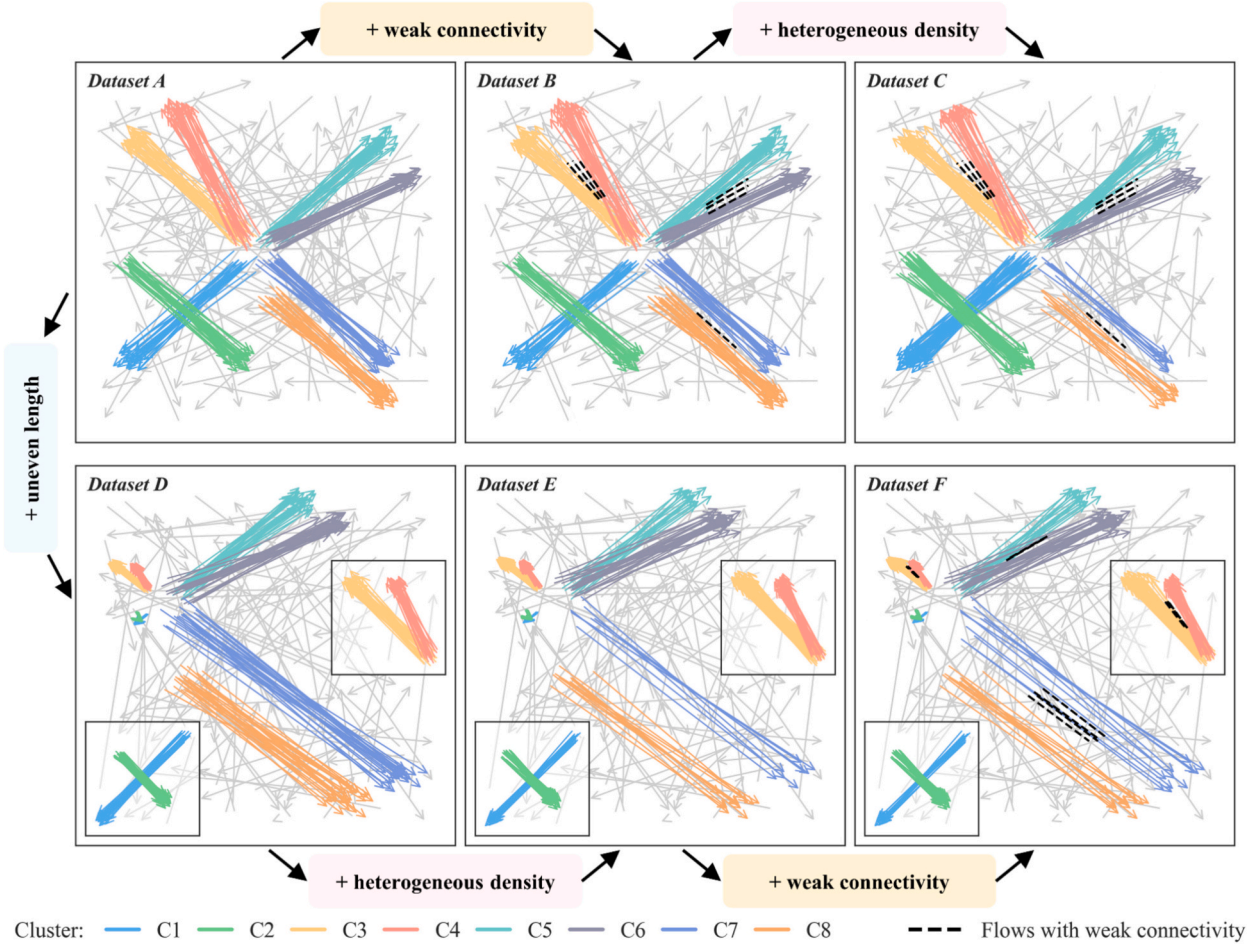


Fig. 5. Six synthetic flow datasets A to F, and their construction procedures.

Fig. 3 (b) (6), the grey-colored flow group is labeled as noise since two flows are insufficient to form a cluster. To be noted, the rule to identify noise flow groups is based on the Definition 5, and can be customized according to application demands. Then, the spatial compactness indicator is calculated for all retained groups to distinguish strongly-connected and weakly-connected flow groups based on the input scale factor. The former is directly preserved as clusters, while the latter undergoes additional processing to detect potential strongly-connected groups in the following step. As shown in Fig. 3 (b) (2), the green-colored group is preserved as a flow cluster, while the orange-colored group needs to be further processed.

3.2.3. *Handling weakly-connected flow groups*

The identified weakly-connected flow group may contain multiple strongly-connected flow groups especially in data with heterogeneous density and weak connectivity. Inspired by the boundary-seeking point clustering algorithms (Peng et al., 2022), this step identifies the cluster boundaries, i.e., partitioning flow (PF), and then separates the flow

Table 2
The highest ARI scores of four flow clustering algorithms on six synthetic datasets.

Dataset	AFC	FlowLF	FlowDBSCAN	ScaleFC	ScaleFC (Adapted)
A	0.818 $k = 44$	**1.000** $r = 3$	**1.000** $\epsilon = 2.5, mf = 5$	**1.000** $\alpha = 0.2, mf = 5$	0.980 $\alpha = 0.2, mf = 7$
B	0.775 $k = 48$	0.850 $r = 2$	0.857 $\epsilon = 1.6, mf = 5$	**0.980** $\alpha = 0.2, mf = 9$	0.889 $\alpha = 0.2, mf = 7$
C	0.763 $k = 52$	0.856 $r = 3$	0.853 $\epsilon = 2.3, mf = 6$	**0.921** $\alpha = 0.19, mf = 5$	0.871 $\alpha = 0.23, mf = 5$
D	0.756 $k = 29$	0.853 $r = 6$	0.901 $\epsilon = 4.5, mf = 5$	**1.000** $\alpha = 0.24, mf = 5$	0.989 $\alpha = 0.24, mf = 8$
E	0.824 $k = 34$	0.814 $r = 6.5$	0.827 $\epsilon = 2.8, mf = 5$	**0.982** $\alpha = 0.23, mf = 5$	0.901 $\alpha = 0.24, mf = 8$
F	0.681 $k = 38$	0.772 $r = 6$	0.766 $\epsilon = 3.8, mf = 5$	**0.974** $\alpha = 0.21, mf = 5$	0.951 $\alpha = 0.25, mf = 6$

Note: *mf* is short for *MinFlows*. Parameters of ScaleFC (Adapted) are derived via the parameter adaptive method outlined in Section 3.4. The best ARI scores for each dataset are highlighted in bold.

clusters with the constraint of these boundaries. Since the PF always lies in the transitions of flow density, we design a density gradient indicator to capture local density variations among individual flows and identify PF. The concrete process of weakly-connected flow groups consists of four sub-steps.

As shown in Fig. 4 (a), first, we assign an ordering index to each flow for calculating the density gradient indicator in the next step. Given a flow group $F = \{f_1, f_2, \ldots, f_n\}$ and the D points $D_1, D_2, \ldots, D_n$, the procedure is as follows:

(1) Obtain the centroid flow with O point denoted as $\overline{O}$ and its direction denoted as $\overline{\theta}$, and determine the scanner line whose direction is $\overline{\theta}_r = (\overline{\theta} + \pi) \ mod \ 2\pi$.
(2) Calculate the direction of flows $\overline{O}D_1, \overline{O}D_2, \ldots \overline{O}D_n$ and obtain the corresponding direction vector $\theta_1, \theta_2, \ldots \theta_n, \overline{\theta}_r$
(3) Arrange the directions in an ascending order $\theta_{t_1}, \theta_{t_2}, \ldots, \theta_{t_i}, \overline{\theta}_r, \theta_{t_{i+1}}, \ldots, \theta_{t_n}$
(4) Assign the flow sequence $f_{t_{i+1}}, f_{t_{i+2}}, \ldots, f_{t_n}, f_{t_1}, f_{t_2}, \ldots, f_{t_i}$ with indexes $1, 2, \ldots, n$

Second, we find the *k*-nearest-neighbors (KNN) of each flow, and the KNN set of f_i is denoted as $KSet_i$, where the flow number $|KSet_i| = k + 1$ since we include flow f_i itself into $KSet_i$ in this study. In this algorithm, we set *k* equals to *MinFlows* as *MinFlows* defines the minimum number of flow neighbors required for clustering. This setting ensures consistency in the density estimation and also reduce the number of input parameters. Fig. 4 (b) shows the KNN result of a weakly-connected flow group in Fig. 3 (b) (5) with $k = 3$.

Then, we utilize the indicator defined in Eq. (7) to measure local density and spatial compactness of the KNN set of f_i:

$$I_{KSet_i} = \sqrt{\frac{\sum_{j=1}^{k+1} D\left(f_j, \overline{f}_{KSet_i}\right)^2}{k+1}} \tag{8}$$

where $\overline{f}_{KSet_i}$ is the theoretical centroid flow of the set $KSet_i$. After that, the density gradient of flow f_i is calculated:

$$density\ gradient\ of\ f_i = \left|\frac{I_{KSet_{i-1}} - I_{KSet_i}}{I_{KSet_i}}\right| + \left|\frac{I_{KSet_{i+1}} - I_{KSet_i}}{I_{KSet_i}}\right| \tag{9}$$

The flow with the maximum density gradient is identified as the PF of the current weakly-connected flow group, demonstrated in Fig. 4 (c).

Final, we divide the flow group into two subgroups using the identified PF. After obtaining the two subgroups, they are fed into the candidate set of unprocessed flow groups and return to Step 2 (Fig. 4 (d)). The algorithm then selects the next unprocessed flow group and repeats the aforementioned steps.

3.2.4. Reallocating partitioning flows and outputting cluster results

After completing the process of all flow groups, we obtain flow clusters, noise flows, and partitioning flows (PFs). At this point, PFs remain unlabeled. Since they represent the boundaries between flow clusters and may belong to a specific cluster, we need to further testify their belongingness. Specifically, for each PF, it is assigned the label of the nearest flow cluster if adding it to that cluster preserves the strongly-connected group; otherwise, it is labeled as noise (Fig. 3 (b) (7)). Following this step, the final flow clusters are produced.

Reviewing the workflow of ScaleFC, there are three parameters, the scale factor α, maximum analytical flow length *MaxLen* and minimum flow number *MinFlows*. According to Eq. (6), parameter α and *MaxLen* are for calculating the neighborhood range of each flow based on its length, where α serves as an adjustable coefficient, and *MaxLen* is the maximum flow length cutoff. *MinFlows* denotes the minimum number of flows required to form a flow cluster for eliminating noisy flows. As *MaxLen* is an application-dependent parameter, we leave it to user upon their analytical goals. For example, larger value should be assigned to taxi OD flows than bike-sharing OD flows, since taxi OD flows usually have longer travel distances. Therefore, we just provide adaptive parameter estimation method for α and *MinFlows* in the following section. Meanwhile, we do not set *MaxLen* in this study as both the synthetic datasets and real-world bike-sharing OD dataset do not contain extreme long flows.

3.3. Time complexity analysis

The first step of ScaleFC has a time complexity of $O(N^2)$, as the distance between any two flows needs to be calculated to search for flow neighbors. In the second step, each flow group is assumed to contain an average of t flows ($MinFlows < t < N$), so the average number of the groups is $\frac{N}{t}$. When calculating the spatial compactness indicator, t times distance calculations are required between the centroid flow and individual flows for each group. Thus, the time complexity of recognizing the strongly-connected flow group is $O\left(\frac{N}{t} \cdot t\right)$, i.e., $O(N)$. In the third step, the time complexity of calculating the spatial compactness indicator for the KNN set of each flow is $O(t^2)$, as the KNN searching and the indicator calculation for each group both requires $O(t^2)$. For computing the density gradient indicator, the complexity is $O(t)$ for each group. Therefore, the time complexity of handling weakly-connected flow groups is $O\left(\frac{N}{t}\left(2t^2 + t\right)\right)$, i.e., $O(tN)$. In the last step, the time complexity of finding the nearest cluster for each PF is $O(1)$, as the neighbor relationships have already been calculated in the first step. Since the spatial compactness indicators of the temporarily generated flow clusters need to be calculated, the time complexity of the fourth step is approximately $O(p(t+1))$, i.e., $O(pt)$, where p is the number of PFs ($p \ll N$).

In summary, the total time complexity of ScaleFC is $O(N^2 + tN + pt)$, where t represents the average number of flows per group, and p denotes the number of PFs. Consequently, the more flows each cluster contains on average, the higher the total time complexity. When using spatial index to speed up the computation, the complexity of the first step can be reduced to $O(N \log N)$. Therefore, the overall time complexity of the algorithm is $O(N \log N)$ in the best case, and $O(N^2)$ in the worst case. Apart from spatial indexing, the parallel computing can be further

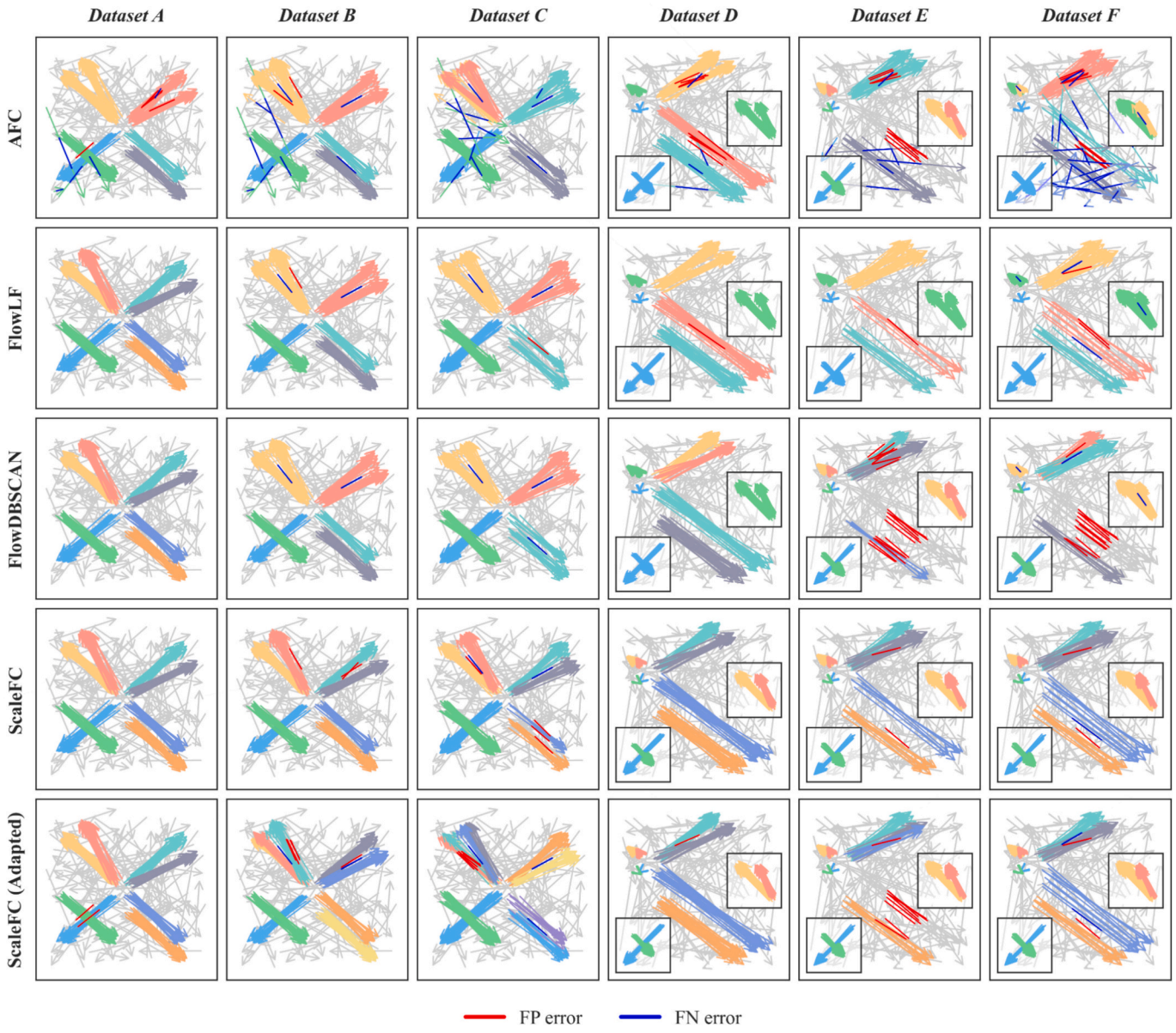


Fig. 6. Clustering results of the four algorithms on six synthetic datasets.

Table 3
Theoretical optimal time complexities of four flow clustering algorithms.

Category	Method	Time complexity
Hierarchical	AFC	$O(kTN^2\log(TN))$
Statistics-based	FlowLF	$O((M+R)N^2\log N)$
Density-based	FlowDBSCAN	$O(N\log N)$
	ScaleFC	$O(N\log N)$

Note: N, k, T are the numbers of flows, nearest flows, and the average number of flow neighbors for all flows respectively; M is the number of observation scales, and R is the iterations of Monte Carlo simulations.

leveraged to accelerate this algorithm since each flow group can be processed independently (Wang et al., 2020).

3.4. Adaptive parameter estimation for MinFlows and scale factor α

Alongside utilizing the brute force searching to determine the two parameters of ScaleFC, data-driven strategies are introduced to obtain their optimal values. First, an appropriate *MinFlows* is determined by analyzing variations in the kth nearest neighbor distance distribution of flows. Then, a suitable α is estimated based on the relation between flow neighborhood range and flow length defined in Eq. (6).

***MinFlows*:** Since each OD flow can be represented as a point in a four-dimensional Cartesian space formed by the O-plane and D-plane (Pei et al., 2020), the method for estimating the number of point events can be adapted for flow data (Liu, Yang, Deng, Song, & Liu, 2022; Tang et al., 2024). Specifically, we estimate the parameter *MinFlows* by a nonparametric index named RKD that captures the variation of the local flow density with different k values (Pei, 2011). The index is defined as follows:

$$RKD = \frac{Var_{k+1}}{Var_k}/R_k \tag{10}$$

where Var_k and Var_{k+1} is the real variance of the kth nearest distance and $(k+1)$th nearest distance of all flows respectively. R_k is a constant that represents the ratio between the expectation of the theoretical

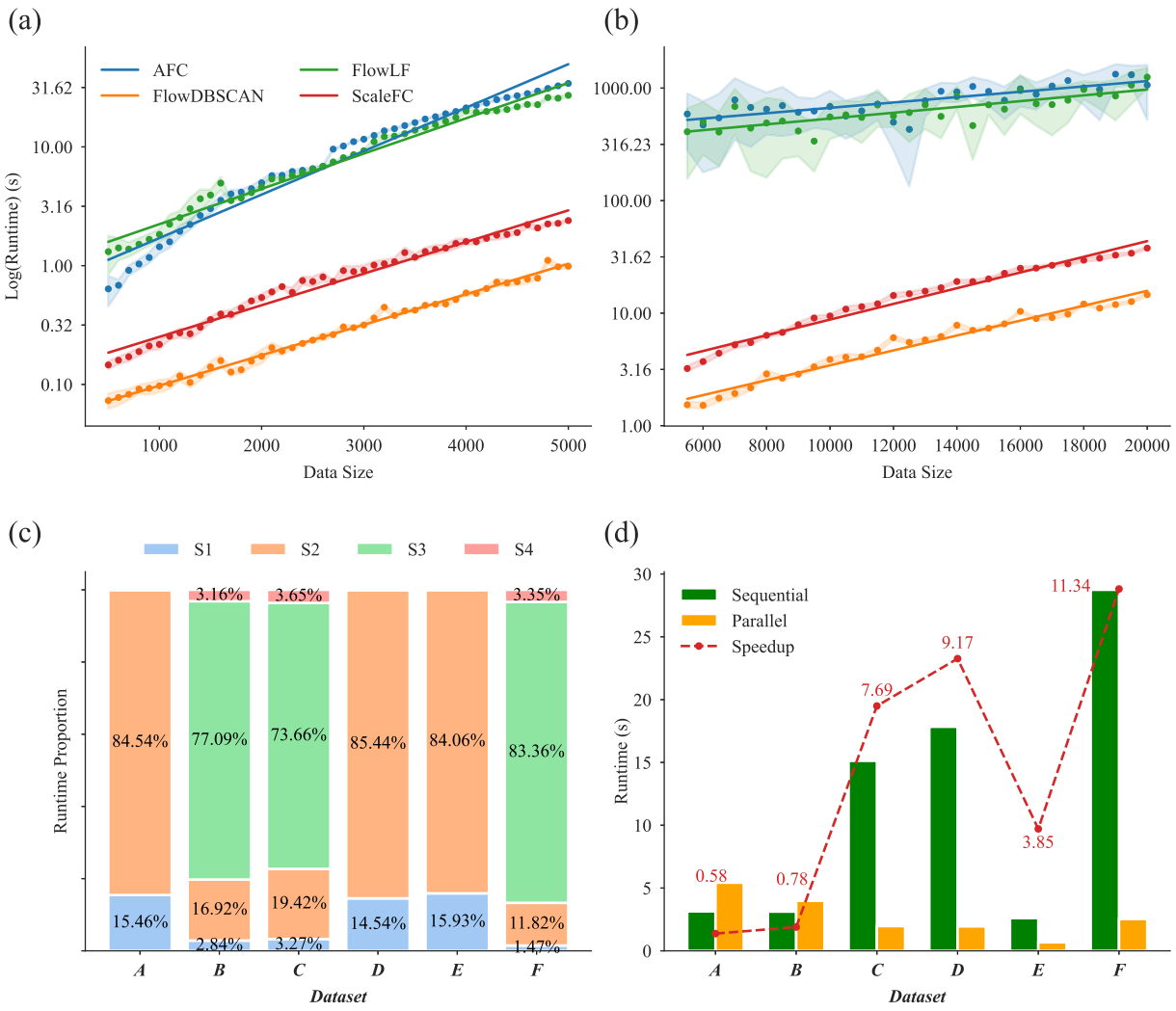


Fig. 7. Runtimes of four flow clustering algorithms on (a) small-scale datasets and (b) large-scale datasets with the increased data size. (c) Stage-wise runtime proportions, (d) parallel and sequential runtimes, and corresponding speedup of ScaleFC across six synthetic datasets.

variance of the $(k+1)$th nearest distance and that of the kth nearest distance for a homogeneous flow set. It can be derived from the homogeneous Poisson process, detailed in Pei (2011).

The RKD value decreases significantly when k increases from between 1 and the optimal value. Once k exceeds the optimal, the RKD stabilizes and presents a minimal increase in variance. Therefore, the estimated *MinFlows* can be identified at the leveling-off change point of the RKD curve. In this study, we employ the second derivative to quantifies the change rate of slope, and choose the point with minimum as change point as it signifies curvature where the change shifts to stable. Such an approach can avoid manual threshold selection (Drăguţ et al., 2010) and unnecessary model assumption (Li et al., 2018).

Scale factor α: Inspired by RKD, we employ a similar strategy to estimate α after determining the appropriate value of *MinFlows*. The first step is to calculate the kth nearest distance (here k is equal to *MinFlows*) for each flow and use this distance as its neighborhood range ϵ. Then, the maximum permissible α for each flow is inferred based on Eq. (6). We arrange the permissible α of all flows within the valid range in an ascending order to generate a curve, and the α at the elbow point is selected as the estimated scale factor. The setting principle is that, for a high-density flow, its kth nearest distance is generally small, resulting in a reduced estimated α. In contrast, a low-density flow or noise flow exhibits larger kth nearest distance, leading to an increased α. Therefore, the elbow point of the scale factor curve can serve as the boundary between feature and noise flows.

4. Validation on synthetic data

To validate the effectiveness of ScaleFC, we designed six synthetic datasets and compared the performance with three state-of-the-art baselines, AFC, FlowLF, and FlowDBSCAN. We also evaluated the computational efficiency of ScaleFC, and analyzed the parameter sensitivity and feasibility of the proposed parameter adaptive setting method of scale factor α and *MinFlows*. Additionally, in Appendix B, we further compared the efficacy of α and a scale-aware flow distance metric, *Flow Dissimilarity* (Tao & Thill, 2016b, 2019a), by integrating them with FlowLF and FlowDBSCAN, respectively. All experiments were conducted on a commodity computer equipped with Windows 11, 16GB of RAM, and a 12th Gen Intel (R) Core (TM) i7-12700H 2.30 GHz processor. The flow clustering algorithms were implemented in Python.

4.1. Synthetic datasets

In the six designed synthetic datasets, ***Dataset A*** is the original dataset whose feature flows have uniform lengths, consistent density, and no weak connectivity between dense clusters. Other five datasets are all derived from ***Dataset A*** by introducing the flow characteristics of uneven length, heterogeneous density, and weak connectivity. These variations can evaluate the impact of each characteristic on flow clustering accuracy and the performance of the clustering algorithm under different cases. In addition, other factors such as noise proportion,

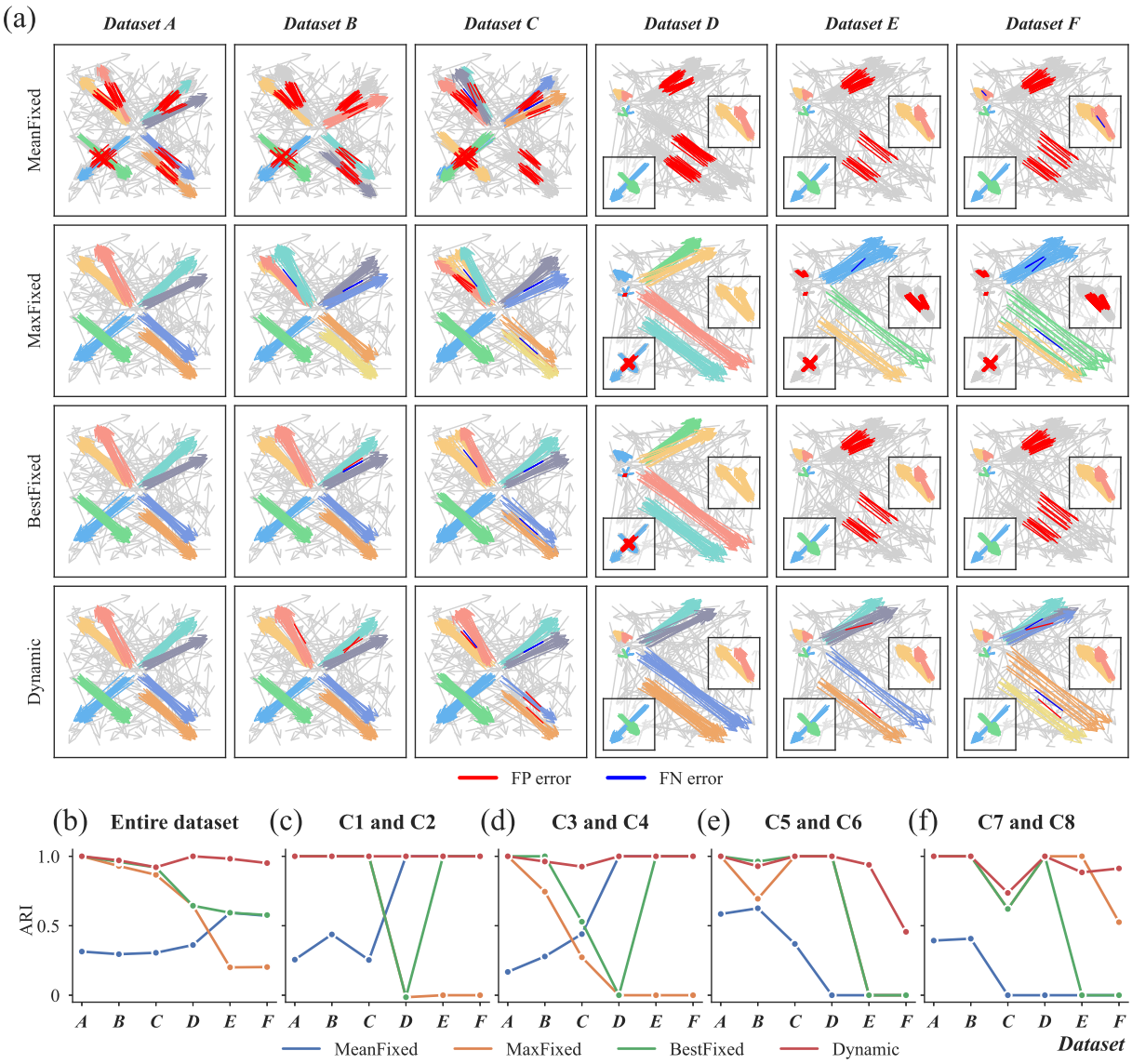


Fig. 8. (a) Clustering results obtained by ScaleFC with fixed and dynamic neighborhoods. ARI scores of (b) the entire dataset, (c) cluster C1 and C2, (d) C3 and C4, (e) C5 and C6, and (f) C7 and C8 respectively. MeanFixed and MaxFixed refer to setting a fixed ϵ-neighborhood using the average and maximum of kth nearest distance values of all flows, respectively; BestFixed employs the fixed ϵ at which the ARI score is highest; while Dynamic adjusts the ϵ based on the scale factor.

relative spatial distribution, and shapes of clusters remain unchanged to reduce their impacts on clustering results. As shown in Fig. 5, each dataset has 100 noise flows, and 200 feature flows contained in eight clusters, i.e., C1 to C8. Flows exhibiting weak connectivity between clusters are red dotted in the central of the flow, while the color of two ends represent their labels. For datasets ***D***, ***E***, and ***F***, the insets in the lower-left corner and the right provide zoomed-in view of clusters C1 and C2, and clusters C3 and C4, respectively. Specifically, ***Dataset B*** introduces flows with weak connectivity in the intermediate regions between two adjacent high-density clusters, namely C3 and C4, C5 and C6, and C7 and C8. ***Dataset C*** contains sparse flow clusters by removing partial flows from C7 and C8 in ***Dataset B***, while an equal number of flows are added to clusters C1 and C2. ***Dataset D*** shortens the lengths of all flow in clusters C1 to C4, and extents lengths of all flow in clusters C5 to C8 respectively. ***Dataset E*** further reduces the density of clusters C7 and C8 to generate sparse clusters, while increasing the density of clusters C5 and C6. Similar to ***Dataset B***, ***Dataset F*** introduces the weak connectivity based on ***Dataset E*** by inserting flows between adjacent high-density clusters.

4.2. Baselines, evaluation metric and parameter settings

We chose the hierarchical algorithm AFC, the statistics-based algorithm FlowLF, and the density-based algorithm FlowDBSCAN, as baselines for comparison. AFC can uncover hierarchical relationships in flow data without a predefined number of clusters. FlowLF provides robust statistical analysis of flow cluster patterns, helping to detect deviations from randomness. FlowDBSCAN is particularly effective at identifying clusters of arbitrary shapes and eliminating noise. In summary, these three methods provide different perspectives for evaluating flow clustering performance.

Specifically, AFC adopts agglomerative approach to combine clusters based on average linkage strategy. The key parameter k is the number of nearest neighbors of flows for calculating SNN similarity. The method adaptively estimates k using the kth nearest distance, and constraints such as 95 % of flows have at least one neighbor and 70 % of flows have at least seven neighbors according to SNN similarity. FlowLF extends the point-based L-function to flow space by defining a flow L-function to determine the best aggregation scale r of flows. The optimal scale is identified as the first minimum of the L-function derivative after its

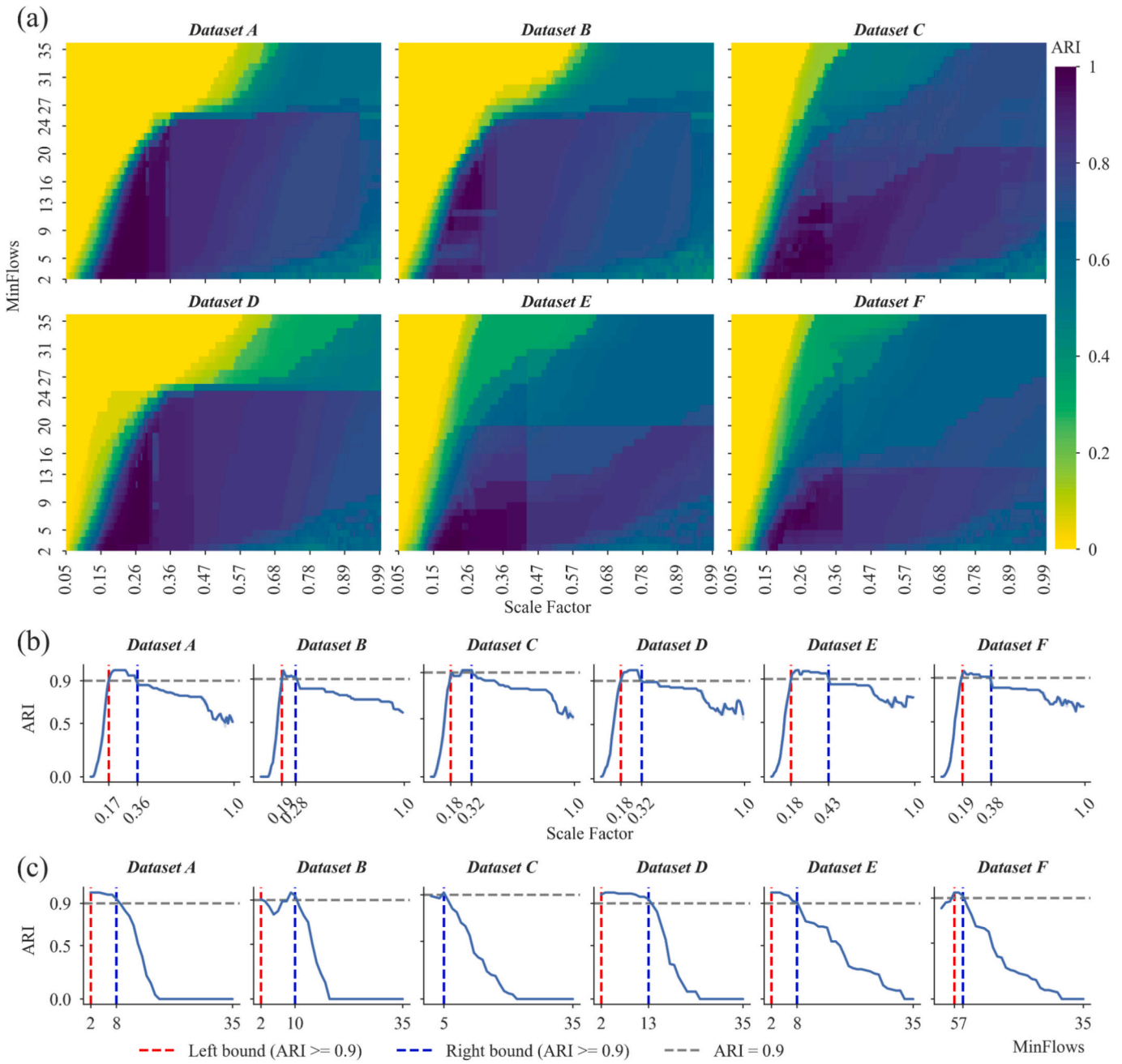


Fig. 9. (a) Trends of ARI score by varying α and *MinFlows* on Dataset A-F. (b) Trends of ARI by varying α but fixing *MinFlows*, and (c) by varying *MinFlows* but fixing α respectively. The range of α and *MinFlows* where the score exceeds 0.9 are highlighted using vertical dotted lines.

global maximum. Using this observation radius, FlowLF computes local L-functions and employs Monte Carlo simulations for statistical significance testing, instead of using a predefined ratio to extract high-density flow clusters. As FlowLF identifies regions with aggregated flows without spatial partitioning, we use a distance-based connection mechanism to form clusters. FlowDBSCAN, inspired by the DBSCAN, first constructs a distance matrix, then identifies core flows, boundary flows, and noise flows based on the input parameters ϵ and *MinFlows*, and generates flow clusters using a density connection mechanism. To ensure fairness in the experimental comparison, the parameter spaces for the four algorithms are as closely aligned as possible. The search increment for the parameter k in AFC, r in FlowLF, ϵ and *MinFlows* in FlowDBSCAN, and α and *MinFlows* in our ScaleFC are 1, 0.01, 0.01 and 1, and 0.01 and 1 respectively. The search spaces are determined based on the distribution of the synthetic datasets, with the following ranges 2–35, 0.01–5, 0.01–5, 2–35, 0.01–1 and 2–35, respectively.

To evaluate the clustering accuracy, an external clustering validity metric, Adjusted Rand Index (ARI) (Hubert & Arabie, 1985), is adopted:

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right] \Big/ \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right] - \left[\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right] \Big/ \binom{n}{2}} \tag{11}$$

where a_i and b_j denote the number of flows in the true cluster i and the predicted cluster j, n_{ij} is the number of flows in the intersection of cluster i and cluster j. ARI ranges from -1 to 1, and the higher ARI, the better clustering performance.

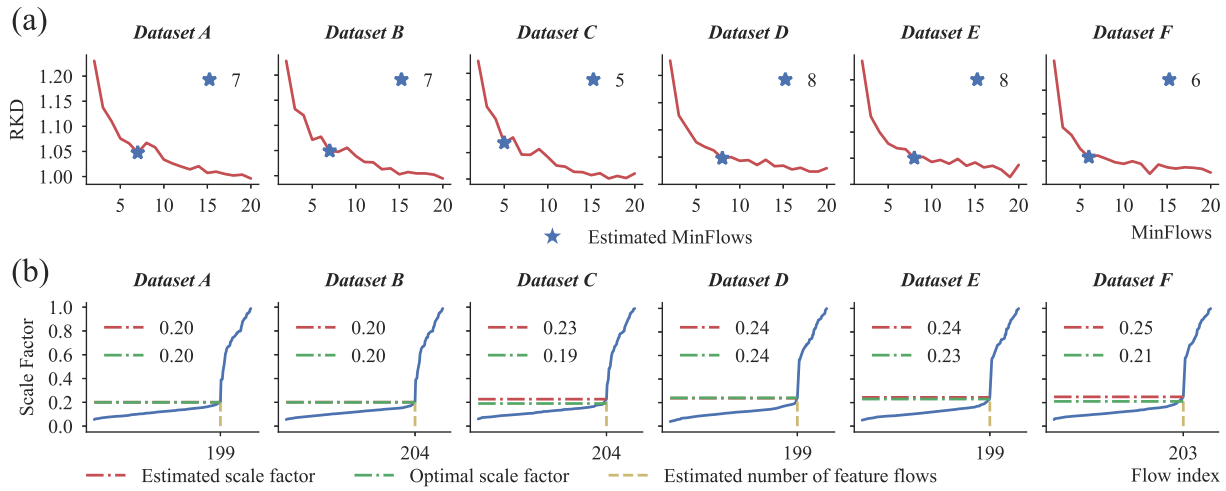


Fig. 10. The results of adaptive parameters. (a) *MinFlows* based on RKD index and (b) scale factor α based on kth nearest distance strategy on six datasets.

4.3. Clustering result comparison

Table 2 shows the highest ARI scores and corresponding parameters of the four algorithms, as well as the results derived from the parameter adaptive method in Section 3.4, denoted as ScaleFC (Adapted), on six synthetic datasets. In general, ScaleFC obtained the best clustering performance, with notable improvements in the ARI scores. It yields at least 10 %, 13 % and 20 % improvements in accuracy over the second-best method under the optimal parameter settings for flows with uneven length (***Dataset D***), flows with heterogeneous density and weak connectivity (***Dataset C***), and flows exhibiting all three characteristics (***Dataset F***) respectively. Although the performance of ScaleFC (Adapted) is slightly inferior to ScaleFC with optimal parameters, it outperforms other three baselines, demonstrating the effectiveness of the adaptive parameter approach. Fig. 6 illustrates the clustering results of the four algorithms on each synthetic dataset, along with a detailed presentation of clustering errors. Red-colored flows represent the false positive (FP) errors, while blue-colored flows denote the false negative (FN) errors. Here, FP errors occur when noise flows are mistakenly classified as feature flows, while FN errors mean that feature flows are misclassified as noise flows.

For ***Dataset A***, FlowLF, FlowDBSCAN, and ScaleFC achieved comparable results, each with an ARI score of 1. However, AFC incorrectly merged the flow clusters C3 and C4, as well as C5 and C6, into single clusters, and misclassified noise flows as part of cluster C2 (Fig. 6). For ***Dataset B***, all the three competitors erroneously merged the two flow clusters, C3 and C4, and C5 and C6, due to their distance-based connectivity mechanisms. In comparison, ScaleFC can distinguish the two high-density clusters with weak connectivity. For ***Dataset C***, FlowLF and FlowDBSCAN incorrectly merged the two sparse clusters C7 and C8 as one, while AFC made similar mistakes across all adjacent clusters. In contrast, ScaleFC yielded a promising clustering result with only 4 FP errors and 2 FN errors. For ***Dataset D***, although FlowDBSCAN distinguished the two clusters C5 and C6, all the three baselines merged the shorter clusters C1 and C2 into a single cluster, despite a nearly 90-degree directional difference between them. This occurs because the use of a fixed larger ϵ-neighborhood caused two overlapped short-distance flows to be regarded as neighbors of each other. Besides, AFC exhibited higher sensitivity to the noise flows. By contrast, ScaleFC achieved an ARI score of 1, demonstrating its effectiveness to handle flows with uneven length. For ***Dataset E***, FlowDBSCAN separated shorter weakly-connected flow clusters by choosing a smaller ϵ, but it misidentified the sparse clusters C7 and C8 as noise. FlowLF selected a larger observation radius, thereby causing flow clusters C1 and C2, C3 and C4, and C5 and C6 to be merged as ones respectively. In comparison, ScaleFC maintained high clustering accuracy in the heterogeneous density flow dataset. For ***Dataset F***, AFC cannot detect the noise flows effectively, while FlowDBSCAN generated overestimated results that incorrectly merged clusters C3 and C4, and underestimated results that considered cluster C7 as noise. ScaleFC performed the best, indicating its advantage in processing flow data with complex patterns.

In summary, AFC is sensitive to noise flows. Even though FlowLF and FlowDBSCAN could handle noise, they cannot separate nearby flow clusters with varying density and weak connectivity. In contrast, ScaleFC can handle flow data with uneven length, heterogeneous density and weak connectivity through dynamic neighborhood adjustment and partitioning flow detection.

4.4. Computational efficiency analysis

Table 3 presents the theoretical optimal time complexities of four algorithms. As a hierarchical clustering method, AFC exhibits the highest complexity. The time complexity of FlowLF is related to the number of Monte Carlo simulations, which requires the calculation of the distance matrix between flows. By utilizing spatial index structures such as R-trees, the time complexities of both FlowDBSCAN and the proposed ScaleFC can be reduced to $O(N\log N)$.

To analyze the computational efficiency, we generated two-scale OD flow datasets with varying sizes, ranging from 500 to 5000 with an interval of 100, and from 5000 to 20,000 with an interval 500 respectively. Each dataset contains a fixed 70 % proportion of feature flows. We ran each algorithm 10 times and present the average runtimes. Fig. 7 (a) shows that AFC has the longest runtime, followed by FlowLF. Although ScaleFC and FlowDBSCAN have the same theoretical time complexity, FlowDBSCAN demonstrates shorter runtime. This is because FlowDBSCAN is implemented upon DBSCAN function provided by sklearn library, which includes optimizations such as efficient neighborhood queries using KD-trees or ball trees, vectorized operations, use of Cython, and parallelization (Pedregosa et al., 2011). Fig. 7 (b) further demonstrates the advantageous computability of ScaleFC for large-scale datasets. For example, the runtime of ScaleFC is approximately 37 and 28 times faster than that of AFC and FlowLF when the dataset size reaches 20,000.

Furthermore, the time distribution across the four stages of ScaleFC was analyzed. Fig. 7 (c) illustrates that stages S1 and S4 have minimal impact on overall runtime of ScaleFC. For datasets without weak connectivity (***A***, ***D***, and ***E***), stage S2 accounts for about 85 % of runtime. In contrast, for datasets with this characteristic (***B***, ***C***, and ***F***), stage S3 dominates, representing over 70 % of runtime. This difference arises because S2 only requires the spatial compactness indicator for the whole

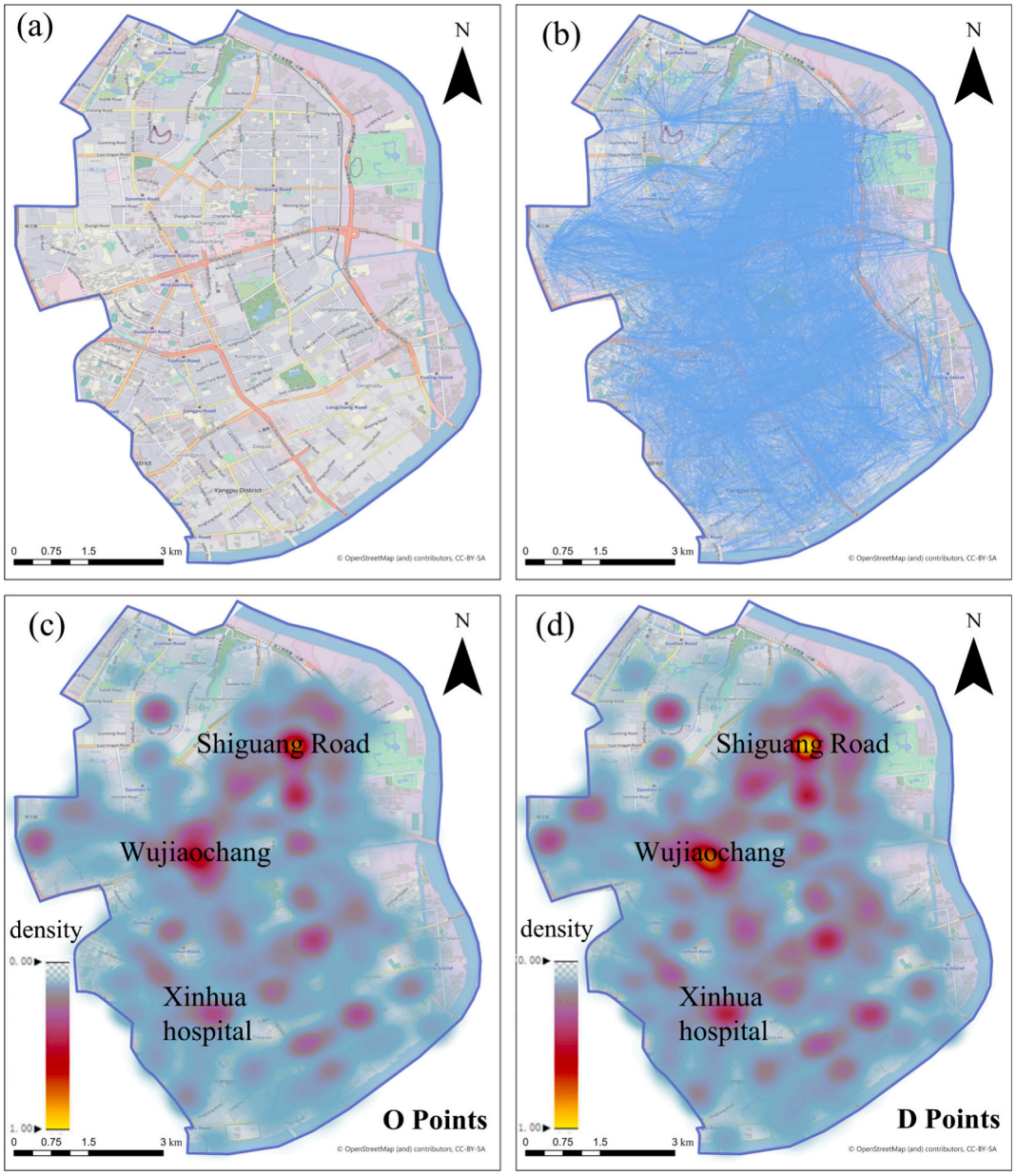


Fig. 11. (a) Yangpu District, Shanghai City, (b) bike-sharing OD flow data in the district, and the heatmap of (c) O and (d) D points respectively.

Table 4
Length statistics of the selected bike-sharing OD flow dataset.

Length (km)	[0.5, 0.75)	[0.75, 1)	[1, 1.5)	[1.5, 2)	[2,3)	[3, 5)	[5, 10]
Number	3331	3213	4378	2083	2119	915	101
Ratio	20.64 %	19.91 %	27.13 %	12.91 %	13.13 %	5.67 %	0.63 %

group, while S3 involves finding the KNN set and calculating the same indicator for each flow in the group. Besides, the independent processing of flow groups in ScaleFC enhances computational efficiency through parallelization. We parallelize the algorithm using multi-processing techniques. As shown in Fig. 7 (d), ***Dataset F*** benefits most, achieving a speedup of 11.34 due to its complex flow patterns. In contrast, datasets ***A*** and ***B*** show speedup below 1, indicating decreased performance due to simpler flow distributions where the overhead of parallelization outweighs its benefits. In summary, ScaleFC exhibits strong computational efficiency and is competent to handle city-level flow datasets.

4.5. Parameter analysis

4.5.1. Effectiveness of scale factor α

ScaleFC utilizes a dynamic neighborhood range based on the scale factor α to identify the neighbors of each flow. To evaluate the effectiveness of α, we compared the clustering performance with the version using fixed neighborhood on six synthetic datasets. We adopted the same settings in Table 2 for *MinFlows*, which are also used to calculate kth nearest distance of each flow. Fig. 8 presents the clustering results of ScaleFC, and ARI scores for the entire datasets and individual clusters using fixed and dynamic neighborhood. In Fig. 8 (b), when the spatial structure of flow dataset becomes complex, ScaleFC with BestFixed and MaxFixed achieve a decreasing clustering performance while MeanFixed showes an upward trend. This can be attributed to the presence of flows with significant length and density variations in the data.

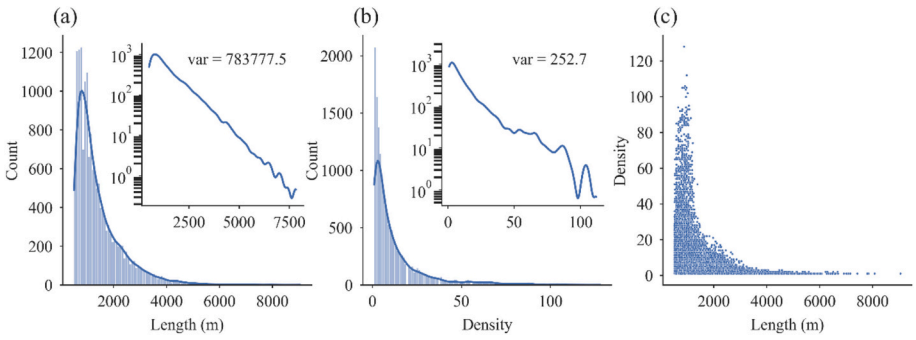


Fig. 12. The (a) length, (b) density histogram, and (c) length-density distribution of the selected bike-sharing OD flows.

Table 5
Three internal evaluation metrics of clustering results and the corresponding parameter settings of four algorithms.

Algorithm	SC ↑	CHI ↑	DBI ↓	Parameter settings
AFC	0.715	5466.370	0.608	$k = 150$
FlowLF	0.691	**5527.620**	0.723	$r = 250(m)$
FlowDBSCAN	0.737	5046.446	0.622	$\epsilon = 300(m), MinFlows = 30$
ScaleFC	**0.795**	5144.244	**0.424**	$\alpha = 0.4, MinFlows = 20$

Note: The best SC, CHI and DBI scores are highlighted in bold.

MaxFixed that utilizes a larger ϵ-neighborhood to identify flow neighbors, tends to misidentify shorter noise flows as feature flows. MeanFixed adopts a balanced ϵ, making it resilient against complex flow patterns. However, despite the upward trend, the ARI of MeanFixed never exceeds 0.6. In contrast, the clustering accuracy with the dynamic neighborhood keeps a high level and is significantly superior to these of the fixed neighborhood. For ***Dataset A***, ***B***, and ***C***, the ARI scores for BestFixed and Dynamic are nearly identical. However, when uneven length is introduced (***Dataset D***), the ARI score for Dynamic improves by 35.5 % compared to BestFixed. When the dataset includes heterogeneous density (***Dataset E***), the ARI score increases by 37.8 %, and with the introduce of weak connectivity (***Dataset F***), it improves by 39.6 %. This improvement validates that the scale factor can adaptively determine the spatial analytical scale of flows, which makes it better capture of local spatial distributions. In comparison, the fixed neighborhood struggles in regions with spatial heterogeneity, generating under- or overestimated clustering results. For example, as shown in Fig. 8 (a), for ***Dataset F***, both MeanFixed and BestFixed, using a smaller ϵ, correctly identify shorter clusters (C1-C4), but misclassify longer clusters (C5-C8) as noise. Figs. 9 (c)-(f) also show that for ***Dataset F***, MeanFixed and BestFixed achieve near-perfect ARI scores for clusters C1-C4, but score 0 for C5-C8. While MaxFixed, using a larger ϵ, identifies C1-C4 as noise, merges C5 and C6 into one cluster, and incorrectly mixes partial flows from C7 and C8 (Fig. 8 (a)).

4.5.2. *Impact of scale factor α and MinFlows*

The impact of two algorithm parameters, scale factor α and *MinFlows*, on clustering accuracy was further evaluated. Fig. 9 (a) illustrates ARI trends for all datasets, where α varies from 0.01 to 1 and *MinFlows* ranges from 2 to 35. The results reveal that all datasets have a wide range of parameter values that yield high clustering results, which demonstrates the robustness of ScaleFC. Figs. 10 (b) and (c) show the effects of varying the α and *MinFlows* on clustering accuracy, respectively. In each experiment, when one parameter is adjusted, the other remains fixed and is set according to Table 2. The parameter ranges that produce an ARI score above 0.9 are highlights.

The results in Fig. 9 (b) show that the ARI score first grows rapidly and then slowly decreases as α increases. When α is small, the neighborhood of each flow is small, which makes longer feature flows to be lost, thereby leading to increasing FN errors in clustering results. While, when α is too large, the neighborhood of each flow expands, causing noise flows to be incorrectly identified as feature flows. In general, ScaleFC is robust to α as it maintains high clustering accuracy within a relatively wide interval. For example, the range of α to achieves an ARI over 0.9 is 0.19, 0.09, 0.14, 0.14, 0.25, 0.19 for dataset A-F, respectively.

MinFlows represents the minimum number of flows in a cluster and is also used for the KNN search in this study. Fig. 9 (c) shows that the optimal *MinFlows* lies between 2 and 8 for these datasets. When *MinFlows* exceeds the range, the clustering accuracy declines rapidly as *MinFlows* increases and eventually drops to zero. Since a higher *MinFlows* value introduces more noise flows, ScaleFC fails to identify flow groups and marks all flows as noise when the parameter is over large. Additionally, compared to the other four datasets, ***Dataset C*** and ***F*** exhibit a narrower range of *MinFlows* required to achieve an ARI above 0.9. This may be attributed to the more complex spatial structure of these datasets, which increase the sensitivity of *MinFlows*. Nevertheless, ScaleFC demonstrates robustness against variations in *MinFlows* in a certain range.

4.5.3. *Adaptive analyses of scale factor α and MinFlow*

We estimated the two parameters using the adaptive parameter estimation outlined in Section 3.4. Fig. 10 (a) illustrates the estimated *MinFlows* derived from the RKD index on six datasets, which all fall within the range of 2 to 8. As evidenced by the analysis in Fig. 9 (c), applying these estimated *MinFlows* values can yield ARIs beyond 0.9. Fig. 10 (b) shows the differences between the optimal α (red dotted line) and the estimated α (green dotted line) using the kth nearest distance on six datasets. The yellow dotted line indicates the number of feature flows satisfying the criteria under the currently estimated α. It shows that the estimated α are generally close to the optimal values, with the number of estimated feature flows consistently approaching 200. These findings support the efficacy of the kth nearest distance-based strategy for determining the α adaptively.

5. Case study on bike-sharing OD data

Bike-sharing OD data captures travel behavior between different locations. Using flow clustering methods to analyze this type of data can identify residential hotspots, commuting routes, and crowd mobility patterns. It provides valuable insights for transportation planning and urban infrastructure optimization (Liu, Gui, et al., 2024; Liu, Li, et al., 2024). To evaluate the applicability of ScaleFC in uncovering urban human mobility patterns, we applied the algorithm to a real-world bike-sharing OD flow dataset.

5.1. *Data and study area*

The bike-sharing OD flow data was collected from Yangpu District, Shanghai City, spanning the period from August 1 to August 31, 2016. Fig. 11 provides an overview of the flow data in the study area, with heatmaps of O and D points respectively. The distribution of hotspots in

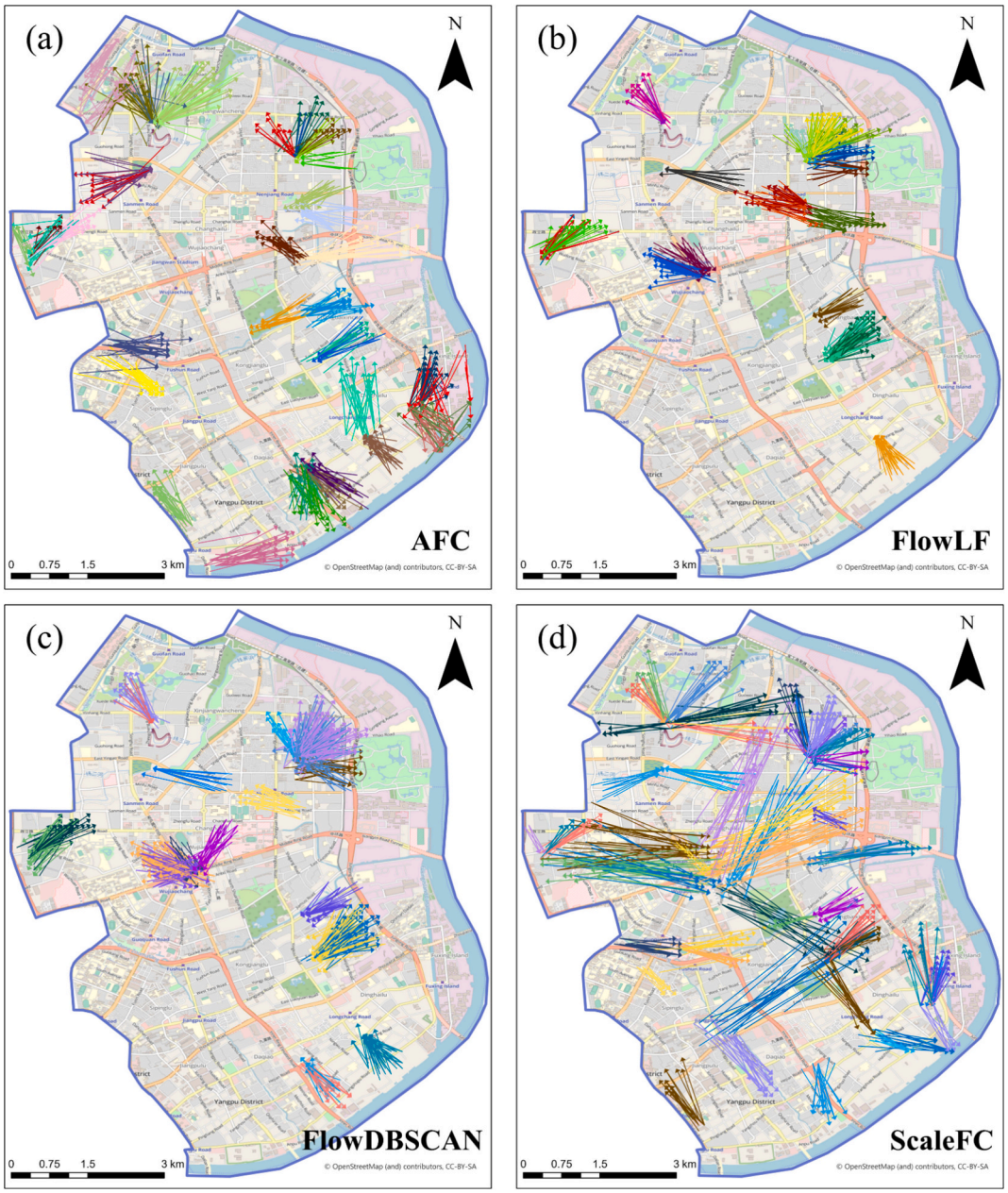


Fig. 13. Comparison of four clustering algorithms on bike-sharing OD flow data.

both heatmaps remains consistent, where Wujiaochang and Shiguang Road identified as two significant aggregation regions. The consistency may be attributed to the strictly designated parking locations in Shanghai City. The dataset includes six attribute fields, i.e., longitude, latitude, and timestamp of both the O and D points. The original spatial reference system, WGS84, was converted to Web Mercator projected coordinates system for clustering calculation. Flows longer than 10 km or shorter than 0.5 km were filtered out due to their infrequent occurrence in the dataset and deviation from typical bike-sharing usage patterns (He et al., 2018). As a result, a total of 16,140 OD flows were retained for analysis.

Before applying clustering algorithms, we analyze the length distribution of the OD flows in Table 4. It is found that 16,039 (99.37 %) flows are under 5 km, with 10,922 (67.68 %) flows fall between 0.5 and 1.5 km, 4202 (26.04 %) flows range from 1.5 to 3 km, and only 1016 (6.3 %) flows extend between 3 and 10 km. This observation is in line with urban bike-sharing travel natures (Kou & Cai, 2019) that bike-sharing is mainly used for short- and medium-distance travel, as shorter trips are often completed on foot, and longer trips are usually made using other transportation modes. Despite about 20.64 % of OD flows being shorter than 750 m, they should not be removed. The length of an OD flow does not necessarily represent the actual trajectory length, and shorter flow clusters may indicate detour travel patterns and potential issues in traffic planning. Besides, the bike-sharing OD dataset does exhibit uneven length characteristic. The number of flows decreases as flow length increases, demonstrating the impact of distance decay effect on flow distribution, where spatial interaction diminishes with increasing distance. Fig. 12 (a)-(b) illustrates the distribution of flow length and density, with density measured based on 500 m neighborhood. It shows that both flow length and density conform to log-linear models, which reflects the spatial heterogeneity and complexity of geographical flows. Fig. 12 (c) further reveals the negative correlation between flow length and flow density. These characteristics make the dataset suitable for evaluating the performance of clustering algorithms in handling uneven

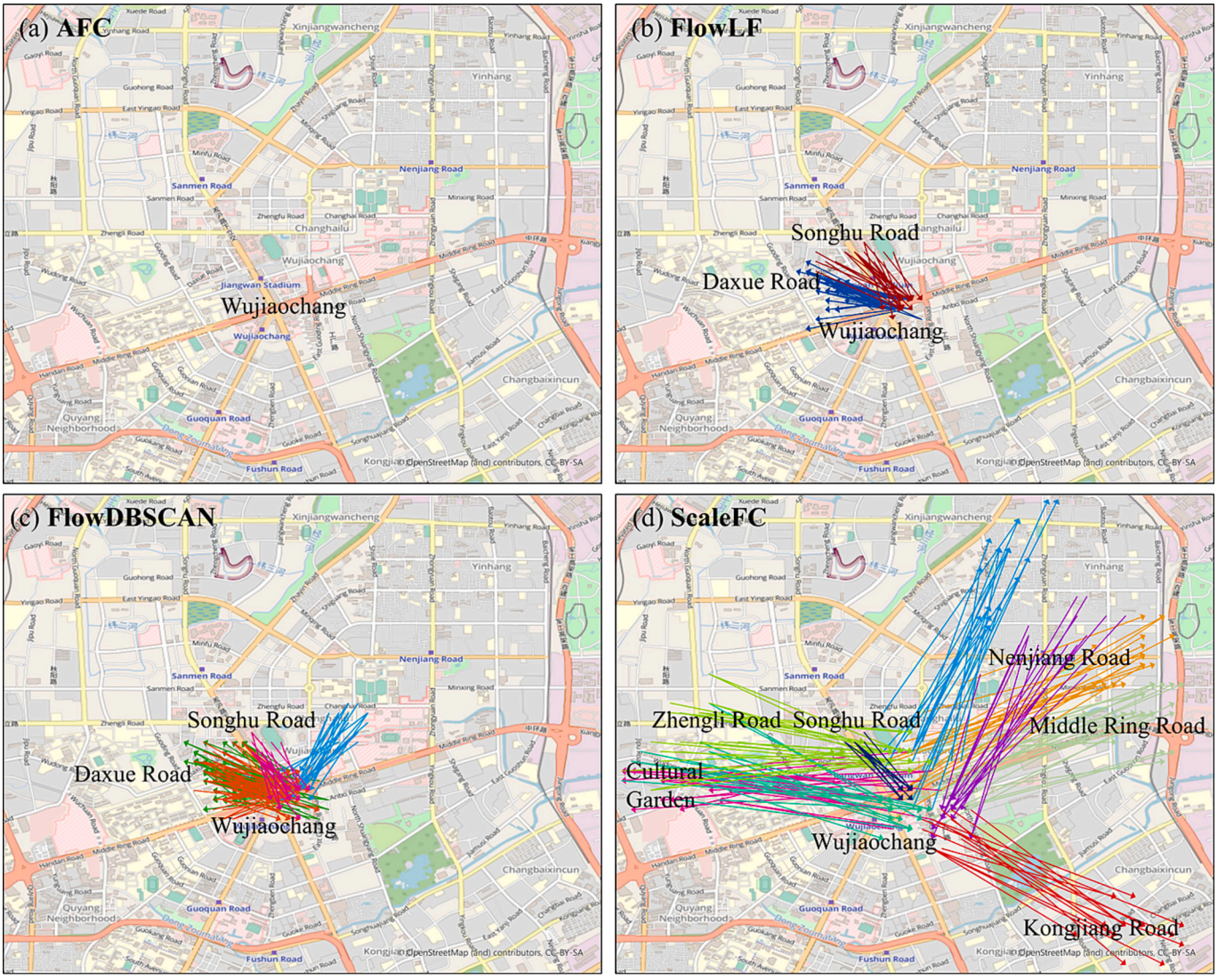


Fig. 14. Comparison of four clustering algorithms around Wujiaochang.

length and heterogeneous density issues.

Since there are no clustering labels for real-world dataset, we adopted three widely-used internal clustering evaluation metrics, i.e., silhouette coefficient (SC) (Rousseeuw, 1987), Calinski-Harabasz Index (CHI) (Caliński & Harabasz, 1974), and Davies-Bouldin Index (DBI) (Davies & Bouldin, 1979), to quantify the clustering performance. SC measures the compactness within clusters and the separation between clusters, CHI assesses the ratio of inter-cluster variance to intra-cluster variance, and DBI evaluates the ratio of intra-cluster compactness to inter-cluster similarity. Integrating these metrics aims to provide a comprehensive assessment of clustering performance, improving evaluation robustness and minimizing potential biases from using a single metric. SC ranges from −1 to 1, with higher values indicating better clustering performance. Both the CHI and DBI range from 0 to positive infinity, with a higher CHI or a lower DBI indicating better clustering results.

5.2. Result analysis

Table 5 displays the results of the three evaluation metrics obtained by four clustering algorithms. The parameters settings for baselines were configured according to the recommended strategies outlined on Section 4.2. ScaleFC achieves the best SC and DBI scores, indicating that the algorithm performs well in terms of both cluster compactness and separation. However, it has a lower CHI score because identifying more clusters leads to reduced variance in inter-cluster distances. Statistically, the number and the flow length variance of clusters generated by AFC, FlowLF, FlowDBSCAN and ScaleFC are 39, 22, 22 and 54, and 70,711.7, 30,057.4, 33,837.6, and 337,787.4, respectively. ScaleFC produces the highest cluster number with highest flow length variance, including both long-distance and short-distance flow clusters. This underscores the capability of our algorithm to capture more abundant flow distribution characteristics. Fig. 13 shows the clustering results of these four algorithms on the maps. The distribution of ScaleFC clusters closely aligns with the heatmap in Fig. 11, which demonstrates its enhanced capability for detecting feature flows. ScaleFC effectively identifies clusters of uneven lengths, ranging from 0.5 km to 2.5 km, and delineates distinct boundaries between clusters. This approach yields clusters with both convergent origins and destinations, concentrating O and D points within smaller areas and indicating fine-scale flow clusters. In comparison, baseline methods miss significant feature flows within hotspots and primarily identify short clusters. For example, AFC fails to detect clusters in the central area Wujiaochang, while FlowLF and FlowDBSCAN ignore clusters nearer the bottom-left corner region Xinhua hospital. Additionally, the baselines produce clusters with convergent-origin and divergent-destination patterns, or vice versa, which do not account for

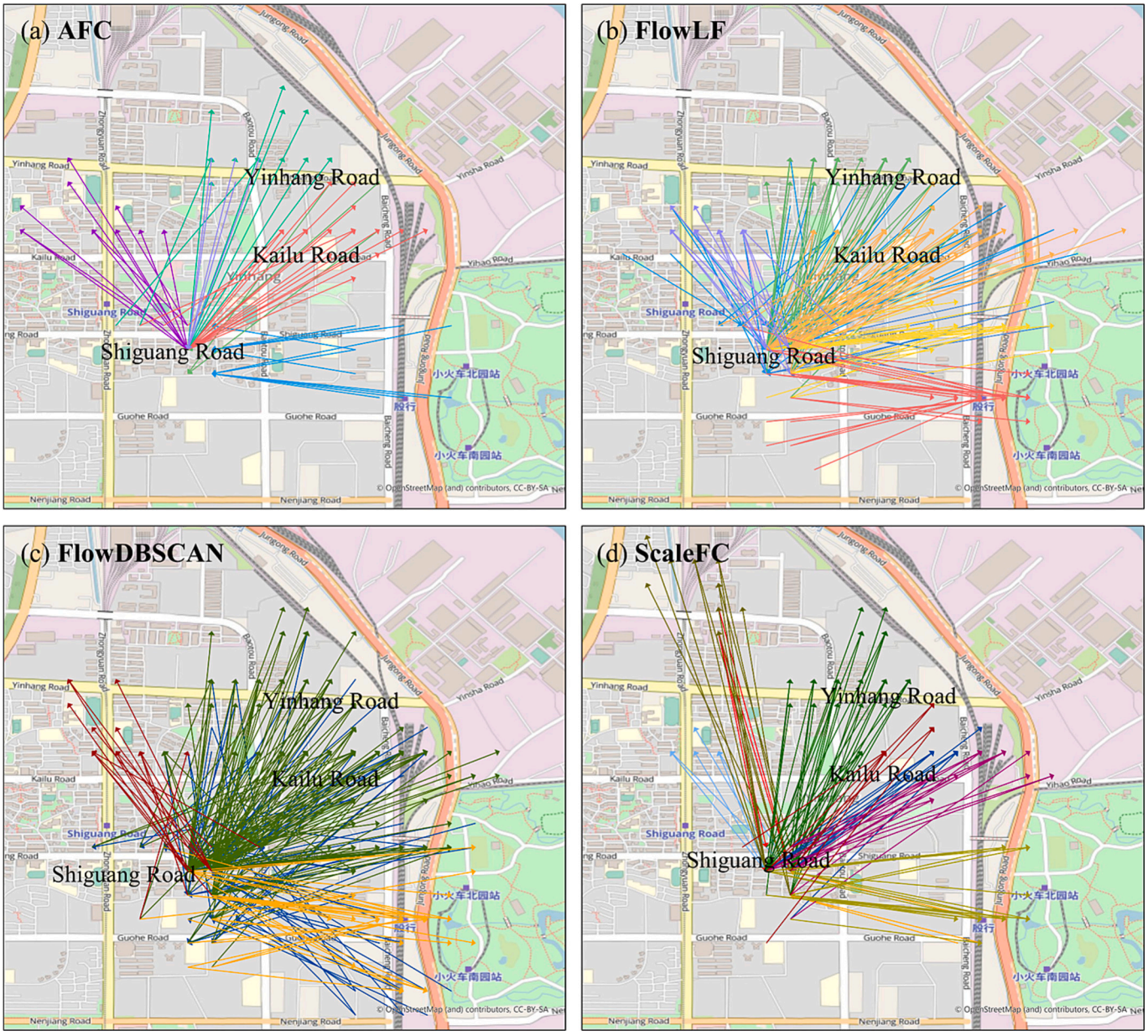


Fig. 15. Comparison of four clustering algorithms around Shiguang Road.

distance-based connectivity and merge separate clusters. In general, the clusters of ScaleFC result in a more accurate and nuanced representation of feature flow clusters than baseline methods.

Nevertheless, Fig. 13 also shows that ScaleFC may ignore certain clusters in local regions, e.g., short clusters detected by FlowLF and FlowDBSCAN around Wujiaochang and a cluster detected by three baselines in the bottom right corner region. The exclusion of these clusters may result from the inconsistent flow clustering scales and the strict definition of strongly-connected flow clusters by ScaleFC. Specifically, the parameter settings of four algorithms are determined according to their respective adaptive parameter methods, making them cluster at different scales. In addition, ScaleFC employs the compactness indicator defined in Eq. (7) to filter out flow clusters. Such a constraint forces ScaleFC to split weakly-connected clusters that fail to meet compactness criteria, thereby compromising mining coarse flow patterns. To preserve more coarse flow patterns, users may either disable flow partitioning or relax compactness constraints. Considering that partitioning flows help to distinguish fine-scale clusters representing distinct patterns, the leverage between pattern diversity and internal consistency is inevitable in concrete application scenarios. Furthermore, the analyses of flow clustering clusters in two prominent hotspots, Wujiaochang and Shiguang Road are detailed.

Wujiaochang, the largest commercial hub in Yangpu District, is well-equipped with a wide range of entertainment, shopping, and dining facilities. Fig. 14 shows the flow clusters identified by four clustering algorithms in the Wujiaochang area. Notably, the AFC algorithm did not detect any clusters. Both FlowLF and FlowDBSCAN identified only two clusters within a radius of approximately 1 km, concentrated along Songhu Road and Daxue Road. In contrast, ScaleFC identified more flow clusters of varying scales, with origins or destinations in nearby streets such as Songhu Road, Zhengli Road, and Cultural Garden, as well as in more distant areas like Nenjiang Road, Middle Ring Road, and Kongjiang Road. ScaleFC also excluded partial smaller clusters in the center of the hub, as their large spatial coverage did not align with the actual flow lengths. The results demonstrates that Wujiaochang area has significant commercial attractiveness with a radius of influence of 2.5 km.

Shiguang Road, situated primarily in a residential area with nearby shops, schools, and hospitals, serves as an informative region for flow clustering analysis. Fig. 15 presents the clustering results in this area. While AFC identified only a limited number of clusters, FlowLF and FlowDBSCAN detected clusters with broader coverage. For example, one cluster originating from Shiguang Road extends across Kailu Road and Yinhang Road, spanning approximately 800 m. In contrast, ScaleFC accurately captured clusters extending in multiple directions from Shiguang Road, which avoids improper merging of distinct clusters. This method is particularly effective in identifying diverse travel patterns within real-world flow data. It facilitates a deeper understanding of travel behaviors across different scales.

6. Conclusion

In this paper, we propose a scale-aware geographical flow clustering algorithm named ScaleFC to identify flow clusters for heterogeneous origin-destination data. We develop a scale factor that adapts with flow length to establish a dynamic neighborhood range for each flow. The scale-aware mechanism can address uneven length issue by allowing a larger neighborhood search range for long flows, and distinguishing overlapped short flows with distinct directions. Meanwhile, we introduce the concept of partitioning flows to obtain the boundaries between adjacent clusters by calculating density gradient indicators for individual flows. These boundaries can be used to separate weakly-connected flow groups to detect potential clusters. Incorporating the two mechanisms also helps to identify clusters with heterogeneous densities. Experimental results on both synthetic datasets and real-world bike-sharing OD data demonstrated that ScaleFC identifies heterogeneous flow cluster patterns more accurately compared to the state-of-the-art algorithms. Specifically, ScaleFC can simultaneously detect both long- and short-distance flow clusters, as well as dense and sparse clusters. The algorithm is able to find human mobility patterns with different travel distance, indicating its potential to provide valuable insights into spatial interactions from a flow-based perspective. Furthermore, the scale factor and partitioning flow mechanisms can be combined with other flow clustering methods to better address issues caused by uneven length, heterogenous density and weak connectivity characteristics.

Nevertheless, the algorithm has limits can be further investigated. Although flow neighborhood range has a positive correlation with flow length, the linear scale factor defined in Eq. (6) may not be the best solution, as spatial associations are non-linear due to scale-dependent and heterogeneous geographical process. Future work could focus on developing a more adaptable model to better capture the relationship between the length and the spatial analytical scales of flows. In addition, the compactness indicator defined in Eq. (7) exhibits limits in processing divergent flow groups with homogeneous densities as discussed in Appendix C. Future research could investigate density variation-based metrics to better handle flows with weak connectivity. Furthermore, the temporal dimension of flow data should also be considered to extend the algorithm to spatiotemporal flow clustering.

CRediT authorship contribution statement

Huan Chen: Writing – review & editing, Writing – original draft, Methodology, Data curation, Conceptualization. **Zhipeng Gui:** Writing – review & editing, Conceptualization. **Dehua Peng:** Writing – review & editing, Methodology. **Yuhang Liu:** Writing – review & editing, Validation. **Yuncheng Ma:** Writing – review & editing. **Huayi Wu:** Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This paper is supported by National Natural Science Foundation of China (No. 42090011, No. 41930107) and Fundamental Research Funds for the Central Universities (No. 2042024kf0005).

Appendix A. Pseudocode of ScaleFC

Algorithm 1: The ScaleFC algorithm
Input: flow data OD, scale factor α, and $MinFlows$
Output: A set of flow clusters $\boldsymbol{SetC}$

Initialize empty sets: feature flows set $SetFF$, noise flows set $SetNF$, flow groups set $SetFG$, strongly-connected flow groups set $SetSG$, weakly-connected flow groups set $SetWG$, and partitioning flows set $SetPFs$

// S1: Identifying flow groups via spatial connectivity
for each flow f in OD **do**
  $fn \leftarrow$ *FindFlowNeighborsByScaleFactor*(f, α)
  if $|fn| \leq MinFlows$ **then**
    $SetNF \leftarrow SetNF \cup \{f\}$
  else
    $SetFF \leftarrow SetFF \cup \{f\}$
  end if
end for
$SetFG \leftarrow$ *MergeFeatureFlowsToFlowGroups*$(SetFF)$ // Do the connection operation
// S2: Recognizing strongly-connected flow groups
for each flow group F in $SetFG$ **do**
  if $|F| \leq MinFlows$ **then**
    $SetNF \leftarrow SetNF \cup F$
  else if *IsStronglyConnectedFlowGroup*(F) **then**
    $SetSG \leftarrow SetSG \cup F$
  else
    $SetWG \leftarrow SetWG \cup F$
  end if
end for
// S3: Handling weakly-connected flow groups
for each flow group F in $SetWG$ **do**
  $F1 \leftarrow$ *AssignIndex*(F) // Get a flow group with index
  $KSet \leftarrow$ *GetEachFlowKNNSet*$(F1, MinFlows)$ // $MinFlows$ is K
  $DG \leftarrow$ *CalculateEachFlowDensityGradient*$(KSet)$
  $PF \leftarrow$ *GetPartitioningFlow*(DG) // Get partitioning flow of current group
  $SetPFs \leftarrow SetPFs \cup \{PF\}$
  $FG1, FG2 \leftarrow$ *SeparateWeaklyConnectedGroup*$(F1, PF)$ // Separate group into two subgroups
  $SetFG \leftarrow SetFG \cup \{FG1,\ FG2\}$
  goto S2
end for
// S4: Reallocating partitioning flows and outputting cluster results
for each partitioning flow PF in $SetPFs$ **do**
  $F \leftarrow$ *FindNearestCluster*$(PF, SetSG)$
  $F1 \leftarrow$ *AddPFToCluster*(PF, F) // add PF to the cluster and get a temporal cluster
  if *IsStronglyConnectedFlowGroup*$(F1)$ **then**
    $F \leftarrow F \cup \{PF\}$ // Add the PF and update the nearest cluster
  else
    $SetNF \leftarrow SetNF \cup \{PF\}$
  end if
end for
$\boldsymbol{SetC} \leftarrow SetSG$
return $\boldsymbol{SetC}$

Appendix B. Evaluation of scale factor and *Flow Dissimilarity*

The *Flow Dissimilarity* (FDS) of two flows f_i and f_j is defined as follows:

$$FDS_{ij} = \sqrt{\frac{a \cdot d_O^2 + b \cdot d_D^2}{L_i L_j}} \tag{B1}$$

where d_O and d_D denote the O-pair and D-pair Euclidean distances, while L_i and L_j denote the length of flow f_i and f_j, respectively; a and b are weight coefficients to control the relative importance of either O or D points (by default $a = b = 1$).

We integrated the scale factor and *Flow Dissimilarity* with FlowLF and FlowDBSCAN, except AFC because it does not require flow neighbor search or flow distance calculation. Table B1 presents the highest ARI scores achieved by each algorithm across six synthetic datasets, along with the parameter settings. Specifically, FlowLF (SF), FlowDBSCAN (SF) and ScaleFC employ a scale factor to identify flow neighbors during the clustering process; FlowLF, FlowDBSCAN, and ScaleFC utilize the flow maximum distance metric; while FlowLF (FDS) and FlowDBSCAN (FDS) employ the *Flow Dissimilarity* metric. Fig. B1 shows the results of the clustering analysis.

Table B1
The highest ARI scores and corresponding parameter settings of three flow clustering algorithms under flow maximum distance, *Flow Dissimilarity*, and our scale factor on six synthetic datasets.

DS	FlowLF	FlowLF(SF)	FlowLF(FDS)	FlowDBSCAN	FlowDBSCAN(SF)	FlowDBSCAN(FDS)	ScaleFC
A	**1.000**	**1.000**	0.979	**1.000**	**1.000**	**1.000**	**1.000**
	$r = 3$	$\alpha = 0.26$	$r = 0.15$	$\epsilon = 2.5$	$\alpha = 0.2$	$\epsilon = 0.1$	$\alpha = 0.2$
				$mf = 5$	$mf = 5$	$mf = 5$	$mf = 5$
B	0.850	0.857	0.850	0.857	0.847	0.843	**0.980**
	$r = 2$	$\alpha = 0.23$	$r = 0.13$	$\epsilon = 1.6$	$\alpha = 0.18$	$\epsilon = 0.09$	$\alpha = 0.2$
				$mf = 5$	$mf = 5$	$mf = 5$	$mf = 9$
C	0.856	0.847	0.862	0.853	0.914	0.899	**0.921**
	$r = 3$	$\alpha = 0.2$	$r = 0.14$	$\epsilon = 2.3$	$\alpha = 0.19$	$\epsilon = 0.1$	$\alpha = 0.19$
				$mf = 6$	$mf = 6$	$mf = 6$	$mf = 5$
D	0.853	0.989	0.980	0.901	0.950	0.989	**1.00**
	$r = 6$	$\alpha = 0.24$	$r = 0.14$	$\epsilon = 4.5$	$\alpha = 0.24$	$\epsilon = 0.11$	$\alpha = 0.24$
				$mf = 5$	$mf = 5$	$mf = 5$	$mf = 5$
E	0.807	0.956	0.931	0.827	0.955	0.973	**0.982**
	$r = 6.5$	$\alpha = 0.16$	$r = 0.12$	$\epsilon = 2.8$	$\alpha = 0.16$	$\epsilon = 0.1$	$\alpha = 0.23$
				$mf = 5$	$mf = 5$	$mf = 5$	$mf = 5$
F	0.772	0.911	0.826	0.766	0.951	0.825	**0.974**
	$r = 6$	$\alpha = 0.17$	$r = 0.1$	$\epsilon = 3.8$	$\alpha = 0.21$	$\epsilon = 0.11$	$\alpha = 0.21$
				$mf = 5$	$mf = 5$	$mf = 5$	$mf = 5$
Rank	4.8	3.0	3.5	4.3	2.8	3.0	**1.0**

Note: *mf* is short for *MinFlows*. The notation (SF) indicates that the algorithm implements the scale factor mechanism for flow neighbor search, while (FDS) denotes that the algorithm employs the *Flow Dissimilarity* metric. The best ARI scores for each dataset are highlighted in bold.

Fig. B1. Clustering results of three flow clustering algorithms under flow maximum distance, *Flow Dissimilarity*, and our scale factor on six synthetic datasets.

The experimental results show that using scale factor can help to address uneven length and heterogeneous density issues for all algorithms. For instance, it improves clustering accuracy by at least 15 % for ***Dataset D*** and ***E*** comparing to these without scale factor. However, incorporating the scale factor cannot solve the weak connectivity issue for FlowLF (SF) and FlowDBSCAN (SF). For instance, for ***Dataset B***, ***C***, and ***F*** with weak connectivity, both the two algorithms exhibit suboptimal performance. For ***Dataset B*** and ***C***, they erroneously merge clusters C3 and C4, and C5 and C6

into one cluster, respectively. For ***Dataset F***, FlowLF (SF) incorrectly merges C5 and C6 into one cluster, while FlowDBSCAN (SF) makes the mistake on the clusters C7 and C8. Table B1 also demonstrates that the *Flow Dissimilarity* metric, can enhance clustering accuracy when handling uneven flow length similar to scale factor; however, it remains insufficient for addressing weak connectivity issue. In addition, the average performance ranks show that ScaleFC achieves the highest position, followed by FlowDBSCAN (SF), FowLF (SF), FlowDBSCAN (FDS) and FlowLF (FDS) with respect to 2.8, 3.0, 3.0 and 3.5. It demonstrates the scale factor performs better than *Flow Dissimilarity* for flow clustering.

In summary, the two proposed mechanisms can be combined with other flow clustering methods. Although scale-aware flow distance metrics such as *Flow Dissimilarity* can achieve effects similar to the scale factor, they modify the distance distribution of flows, making them unsuitable for flow statistical analysis and algorithm parameter estimation.

Appendix C. Applicability and limits of compactness indicator in coping with weak connectivity

The presence of weak connectivity in a flow group should be determined by local density variations. Fig. C1 (a) shows two flow groups with same spatial extents for their respective O and D points distributions, along with their corresponding density variation curves on the right. The density of each flow is calculated by counting flows within identical circle buffers (with a radius of 0.2) around it, with all densities normalized to [0, 1] in a group. *F*1 contains two high-density clusters that are weakly-connected by several low-density flows, showing significant density differences in this group. *F*2 is a uniform-density cluster that has no significant density peaks or valleys.

In this study, we designed a compactness indicator I_F (Eq. (7)) to detect weak connectivity within flow groups. It measures the degree of closeness among all individual flows in a group. To demonstrate the relationship between I_F and flow distribution, we provide an illustration using the toy datasets in Fig. C1 (b). For convenience in discussing the upper and lower bounds of I_F, we assume that each flow group contains n flows where all flows have length of 1, all O points coincide, and D points follow a progressive dispersion pattern on the unit circle. As shown in Fig. C1 (b), I_F reaches its minimum value of 0 when all D points coincide, and achieves its theoretical maximum of 1 when D points are uniformly distributed on the unit circle. The right curve in Fig. C1 (b) further demonstrates that I_F increases progressively as D points distribution changes from convergent to divergent, indicating that I_F can measure the spatial compactness of a flow group.

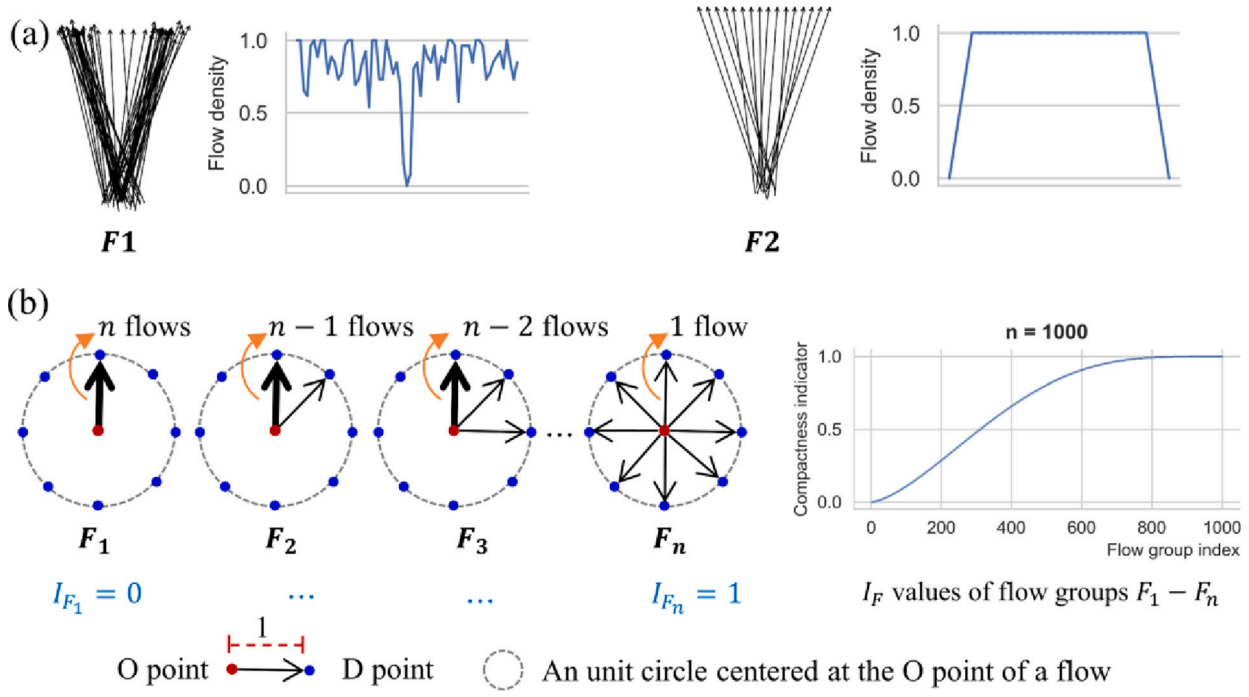


Fig. C1. The illustration of (a) flow groups F1 with weak connectivity and F2 without weak connectivity, and the corresponding local density variations, (b) the compactness indicator of flow groups with different distributions of D points.

I_F can identify weakly-connected groups in most cases, but it may become ineffective for dispersed flow groups with uniform density distribution. In the main text, six synthetic datasets were designed for simulation experiments to validate the effectiveness of I_F in identifying weak connectivity. Fig. 6 in Section 4.3 shows that I_F is able to accurately separate weakly-connected groups when significant density variations exist within the groups. However, when flows are sparsely distributed with uniform densities, I_F tends to be generally high (e.g., F_n in the Fig. C1 (b)). In such cases, these groups are likely to be misclassified as weakly-connected, which contradicts the definition that weakly-connected groups should exhibit local density differences. To illustrate the limitations of I_F, we design two synthetic dataset, ***C-A*** with column-shaped clusters and ***C-B*** with fan-shaped clusters. As shown in Fig. C2, each dataset has 5 clusters C1-C5, and each cluster has 20 feature flows. Flows with weak connectivity are located between clusters C1 and C2, C2 and C3, C3 and C4, respectively. The flows in Cluster C5 have uniform densities, and the cluster has no significant weak connectivity.

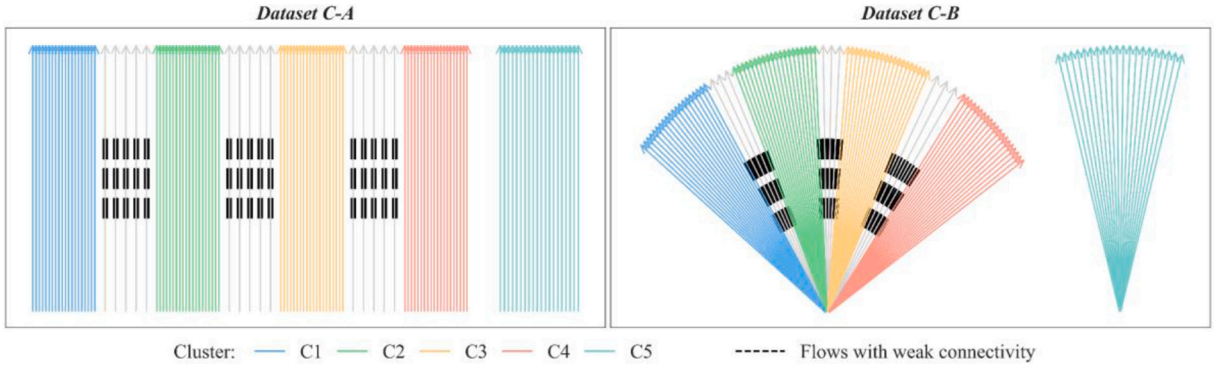


Fig. C2. Synthetic flow datasets with column-shaped clusters (***C-A***) and fan-shaped clusters (***C-B***) respectively.

As illustrated in Fig. C3, ScaleFC copes with weak connectivity and accurately separates clusters C1-C4 in these two datasets. However, it incorrectly splits C5 into two sub-clusters. The algorithm uses I_F to misclassify C5 as weakly connected, and then attempts to detect partitioning flows within the cluster. Since flows in C5 have uniform local densities, all calculated density gradients are zero, making it theoretically impossible to identify a suitable partitioning flow. In our code implementation, the algorithm selects the flow with the maximum gradient based on index ordering, resulting in incorrect separation of cluster C5.

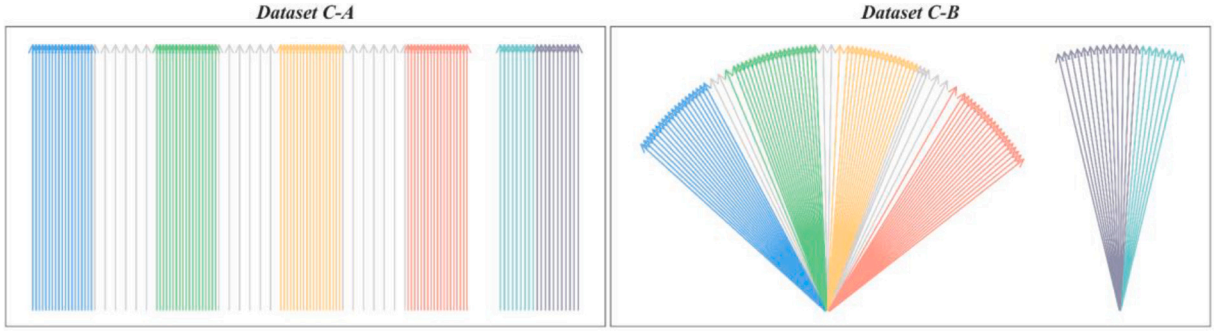


Fig. C3. Best clustering results and the corresponding ARI scores of ScaleFC on two synthetic datasets ***C-A*** and ***C-B***.

In summary, weak connectivity in OD flows depends on density variations rather than absolute density or compactness for a flow group. The indicator I_F works well for flow groups that exhibit significant density differences, as it can use density gradient boundaries to separate weakly-connected clusters. However, I_F may split dispersed flow groups with uniform density into subclusters, causing unexpected results. The reason is that the indicator measures closeness among individual flows within a group but cannot directly quantify density variations. In practice, dispersed flow groups characterized by uniform density are rarely considered as valid clusters at most clustering scales. This is because the distribution of O or D points across extensive spatial areas contradicts the definition of flow cluster pattern, which necessitates the presence of both O and D points within restricted spatial neighborhoods. Thus, I_F remains effective for natural flows in real-world applications. In future, we will investigate density variation-based metrics to better detect flows with weak connectivity, and it may enhance the reliability for identifying diverse flow patterns.

Data availability

Data will be made available on request.

References

Adrienko, N., & Adrienko, G. (2011). Spatial generalization and aggregation of massive movement data. *IEEE Transactions on Visualization and Computer Graphics, 17*(2), 205–219. https://doi.org/10.1109/TVCG.2010.44

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

Anselin, L. (1995). Local indicators of spatial association—LISA. *Geographical Analysis, 27*(2), 93–115. https://doi.org/10.1111/j.1538-4632.1995.tb00338.x

Berglund, S., & Karlstrm, A. (1999). Identifying local spatial association in flow data. *Journal of Geographical Systems, 1*(3), 219–236. https://doi.org/10.1007/s101090050013

Cai, J., & Kwan, M.-P. (2022). Detecting spatial flow outliers in the presence of spatial autocorrelation. *Computers, Environment and Urban Systems, 96*, Article 101833. https://doi.org/10.1016/j.compenvurbsys.2022.101833

Caliński, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics, 3*(1), 1–27. https://doi.org/10.1080/03610927408827101

Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1*(2), 224–227. https://doi.org/10.1109/TPAMI.1979.4766909

Drăguţ, L., Tiede, D., & Levick, S. R. (2010). ESP: A tool to estimate scale parameter for multiresolution image segmentation of remotely sensed data. *International Journal of Geographical Information Science*. https://doi.org/10.1080/13658810903174803

Gao, X., Liu, Y., Yi, D., Qin, J., Qu, S., Huang, Y., & Zhang, J. (2020). A spatial flow clustering method based on the constraint of origin-destination points' location. *IEEE Access, 8*, 216069–216082. https://doi.org/10.1109/ACCESS.2020.3040852

Gao, Y., Li, T., Wang, S., Jeong, M.-H., & Soltani, K. (2018). A multidimensional spatial scan statistics approach to movement pattern comparison. *International Journal of Geographical Information Science, 32*(7), 1304–1325. https://doi.org/10.1080/13658816.2018.1426859

Gui, Z., Peng, D., Wu, H., & Long, X. (2020). MSGC: Multi-scale grid clustering by fusing analytical granularity and visual cognition for detecting hierarchical spatial patterns. *Future Generation Computer Systems, 112*, 1038–1056. https://doi.org/10.1016/j.future.2020.06.053

Guo, D., Zhu, X., Jin, H., Gao, P., & Andris, C. (2012). Discovering spatial patterns in origin-destination mobility data. *Transactions in GIS, 16*(3), 411–429. https://doi.org/10.1111/j.1467-9671.2012.01344.x

Guo, X., Fang, M., Tang, L., Kan, Z., Yang, X., Pei, T., Li, Q., & Li, C. (2025). An adaptive OD flow clustering method to identify heterogeneous urban mobility trends. *Journal*

of Transport Geography, 123, Article 104080. https://doi.org/10.1016/j.jtrangeo.2024.104080

Guo, X., Xu, Z., Zhang, J., Lu, J., & Zhang, H. (2020). An OD flow clustering method based on vector constraints: A case study for Beijing taxi origin-destination data. *ISPRS International Journal of Geo-Information, 9*(2), Article 128. https://doi.org/10.3390/ijgi9020128

He, B., Zhang, Y., Chen, Y., & Gu, Z. (2018). A simple line clustering method for spatial analysis with origin-destination data and its application to bike-sharing movement data. *ISPRS International Journal of Geo-Information, 7*(6), Article 203. https://doi.org/10.3390/ijgi7060203

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification, 2*(1), 193–218. https://doi.org/10.1007/BF01908075

Kou, Z., & Cai, H. (2019). Understanding bike sharing travel patterns: An analysis of trip data from eight cities. *Physica A: Statistical Mechanics and its Applications, 515*, 785–797. https://doi.org/10.1016/j.physa.2018.09.123

Li, F., Gui, Z., Zhang, Z., Peng, D., Tian, S., Yuan, K., Sun, Y., Wu, H., Gong, J., & Lei, Y. (2020). A hierarchical temporal attention-based LSTM encoder-decoder model for individual mobility prediction. *Neurocomputing, 403*, 153–166. https://doi.org/10.1016/j.neucom.2020.03.080

Li, Z., Liu, Q., Tang, J., & Deng, M. (2018). An adaptive method for clustering spatio-temporal events. *Transactions in GIS, 22*(1), 323–347. https://doi.org/10.1111/tgis.12312

Liu, Q., Yang, J., Deng, M., Liu, W., & Xu, R. (2022). BiFlowAMOEBA for the identification of arbitrarily shaped clusters in bivariate flow data. *International Journal of Geographical Information Science, 36*(9), 1784–1808. https://doi.org/10.1080/13658816.2022.2072850

Liu, Q., Yang, J., Deng, M., Song, C., & Liu, W. (2022). SNN_flow: A shared nearest-neighbor-based clustering method for inhomogeneous origin-destination flows. *International Journal of Geographical Information Science, 36*(2), 253–279. https://doi.org/10.1080/13658816.2021.1899184

Liu, Y., Gui, Z., Xu, Y., Gao, S., Zhao, A., Meng, F., Peng, D., Li, F., Bo, L., Wu, H., & Gong, J. (2024). Profiling mobility patterns and driving behaviors of individual drivers via trajectory trait. *The Innovation Geoscience, 3*(1), 100114–100115. https://doi.org/10.59717/j.xinn-geo.2024.100114

Liu, Y., Tong, D., & Liu, X. (2015). Measuring spatial autocorrelation of vectors. *Geographical Analysis, 47*(3), 300–319. https://doi.org/10.1111/gean.12069

Liu, Z., Li, R., Cai, J., Hu, Q., & Wu, H. (2024). Mobility difference index: A quantitative method for detecting human mobility difference. *GIScience & Remote Sensing, 61*(1), Article 2301274. https://doi.org/10.1080/15481603.2023.2301274

Nielsen, T. A. S., & Hovgesen, H. H. (2008). Exploratory mapping of commuter flows in England and Wales. *Journal of Transport Geography, 16*(2), 90–99. https://doi.org/10.1016/j.jtrangeo.2007.04.005

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830.

Pei, T. (2011). A nonparametric index for determining the numbers of events in clusters. *Mathematical Geosciences, 43*(3), 345–362. https://doi.org/10.1007/s11004-011-9325-x

Pei, T., Shu, H., Guo, S., Song, C., Chen, J., Liu, Y., & Wang, X. (2020). The concept and classification of spatial patterns of geographical flow. *Journal of Geo-Information Science, 22*(1), 30–40. https://doi.org/10.12082/dqxxkx.2020.190736

Pei, T., Wang, W., Zhang, H., Ma, T., Du, Y., & Zhou, C. (2015). Density-based clustering for data containing two types of points. *International Journal of Geographical Information Science, 29*(2), 175–193. https://doi.org/10.1080/13658816.2014.955027

Peng, D., Gui, Z., Wang, D., Ma, Y., Huang, Z., Zhou, Y., & Wu, H. (2022). Clustering by measuring local direction centrality for data with heterogeneous density and weak connectivity. *Nature Communications, 13*(1), 5455. https://doi.org/10.1038/s41467-022-33136-9

Ripley, B. D. (1976). The second-order analysis of stationary point processes. *Journal of Applied Probability, 13*(2), 255–266. https://doi.org/10.2307/3212829

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics, 20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Schläpfer, M., Dong, L., O'Keeffe, K., Santi, P., Szell, M., Salat, H., … West, G. B. (2021). The universal visitation law of human mobility. *Nature, 593*(7860), 522–527. https://doi.org/10.1038/s41586-021-03480-9

Shu, H., Pei, T., Song, C., Chen, X., Guo, S., Liu, Y., Chen, J., Wang, X., & Zhou, C. (2021). L-function of geographical flows. *International Journal of Geographical Information Science, 35*(4), 689–716. https://doi.org/10.1080/13658816.2020.1749277

Song, C., Pei, T., Ma, T., Du, Y., Shu, H., Guo, S., & Fan, Z. (2019). Detecting arbitrarily shaped clusters in origin-destination flows using ant colony optimization. *International Journal of Geographical Information Science, 33*(1), 134–154. https://doi.org/10.1080/13658816.2018.1516287

Song, C., Pei, T., & Shu, H. (2020). Identifying flow clusters based on density domain decomposition. *IEEE Access, 8*, 5236–5243. https://doi.org/10.1109/ACCESS.2019.2963107

Tang, J., Zhao, Y., Yang, X., Deng, M., Liu, H., Ding, C., Peng, J., & Mei, X. (2024). Statistical and density-based clustering of geographical flows for crowd movement patterns recognition. *Applied Soft Computing, 163*, Article 111912. https://doi.org/10.1016/j.asoc.2024.111912

Tao, R., Chen, Y., & Thill, J.-C. (2023). A space-time flow LISA approach for panel flow data. *Computers, Environment and Urban Systems, 106*, Article 102042. https://doi.org/10.1016/j.compenvurbsys.2023.102042

Tao, R., & Thill, J.-C. (2016a). A density-based spatial flow cluster detection method. *International Conference on GIScience Short Paper Proceedings, 1*. https://doi.org/10.21433/B3118MF4R9RW

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

Tao, R., & Thill, J.-C. (2019a). Flow cross K-function: A bivariate flow analytical method. *International Journal of Geographical Information Science, 33*(10), 2055–2071. https://doi.org/10.1080/13658816.2019.1608362

Tao, R., & Thill, J.-C. (2019b). flowAMOEBA: Identifying regions of anomalous spatial interactions. *Geographical Analysis, 51*(1), 111–130. https://doi.org/10.1111/gean.12161

Tao, R., Thill, J.-C., Depken, C., & Kashiha, M. (2017). flowHDBSCAN: A hierarchical and density-based spatial flow clustering method. In *Proceedings of the 3rd ACM SIGSPATIAL workshop on smart cities and urban analytics* (pp. 1–8). https://doi.org/10.1145/3152178.3152189

Tran, T. N., Drab, K., & Daszykowski, M. (2013). Revised DBSCAN algorithm to cluster data with dense adjacent clusters. *Chemometrics and Intelligent Laboratory Systems, 120*, 92–96. https://doi.org/10.1016/j.chemolab.2012.11.006

Wang, Y., Gui, Z., Wu, H., Peng, D., Wu, J., & Cui, Z. (2020). Optimizing and accelerating space–time Ripley's K function based on apache spark for distributed spatiotemporal point pattern analysis. *Future Generation Computer Systems, 105*, 96–118. https://doi.org/10.1016/j.future.2019.11.036

Wesolowski, A., Eagle, N., Tatem, A. J., Smith, D. L., Noor, A. M., Snow, R. W., & Buckee, C. O. (2012). Quantifying the impact of human mobility on malaria. *Science, 338*(6104), 267–270. https://doi.org/10.1126/science.1223467

Wood, J., Dykes, J., Slingsby, A., & Radburn, R. (2009). *Flow trees for exploring spatial trajectories.*.

Xiang, Q., & Wu, Q. (2019). Tree-based and optimum cut-based origin-destination flow clustering. *ISPRS International Journal of Geo-Information, 8*(11), Article 477. https://doi.org/10.3390/ijgi8110477

Yan, X., Chen, H., Huang, H., Liu, Q., & Yang, M. (2021). Building typification in map generalization using affinity propagation clustering. *ISPRS International Journal of Geo-Information, 10*(11), Article 11. https://doi.org/10.3390/ijgi10110732

Yan, X., Song, C., Pei, T., Wang, X., Wu, M., Liu, T., … Chen, J. (2022). Revealing spatiotemporal matching patterns between traffic flux and road resources using big geodata-a case study of Beijing. *Cities, 127*, Article 103754. https://doi.org/10.1016/j.cities.2022.103754

Yao, X., Zhu, D., Gao, Y., Wu, L., Zhang, P., & Liu, Y. (2018). A stepwise spatio-temporal flow clustering method for discovering mobility trends. In , *6*. *IEEE Access* (pp. 44666–44675). https://doi.org/10.1109/ACCESS.2018.2864662.

Zhong, X., & Duckham, M. (2016). Characterizing the shapes of noisy, non-uniform, and disconnected point clusters in the plane. *Computers, Environment and Urban Systems, 57*, 48–58. https://doi.org/10.1016/j.compenvurbsys.2016.01.003

Zhou, M., Yang, M., & Chen, Z. (2023). Flow colocation quotient: Measuring bivariate spatial association for flow data. *Computers, Environment and Urban Systems, 99*, Article 101916. https://doi.org/10.1016/j.compenvurbsys.2022.101916

Zhu, X., & Guo, D. (2014). Mapping large spatial flow data with hierarchical clustering. *Transactions in GIS, 18*(3), 421–435. https://doi.org/10.1111/tgis.12100