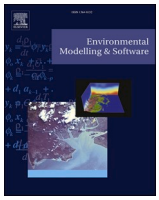
Environmental Modelling and Software 171 (2024) 105865

Contents lists available at ScienceDirect

Environmental Modelling and Software

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



Multiple equidistant belt technique for width function estimation through a two-segmented-distance strategy

Pengfei Wu [a,b], Jintao Liu [a,b,*], Meiyan Feng [a,b], Hu Liu [c]

[a] The National Key Laboratory of Water Disaster Prevention, Hohai University, Nanjing, 210024, China
[b] College of Hydrology and Water Resources, Hohai University, Nanjing, 210024, China
[c] Linze Inland River Basin Research Station, Chinese Ecosystem Research Network, Lanzhou, 730000, China

ARTICLE INFO

Handling Editor: Daniel P Ames

Keywords:
Flow distance
Width function
Two-segmented-distance strategy
Triangulation with linear interpolation
Multiple equidistant belt technique

ABSTRACT

In order to estimate flow distance and width function more accurately, the triangulation with linear interpolation (TLI) method to approximate the missing flow distance values within a cell except for the cell center. Then a new flow distance algorithm (D∞-TLI) is proposed to estimate the flow distance using a two-segment-distance strategy, while two segmented-distances in and outside a local 3 × 3 window are severally modelled with D∞ and TLI. Then, using the continuous flow distance field approximated over a cell region, this cell can be divided into multiple equidistant belts (MEB) to estimate the width function. Assessments using four numerical and two real-world terrains demonstrate that D∞-TLI outperforms nine existing flow distance algorithms and has a mean absolute relative error (*MARE*) lower than 5%. Moreover, the combination of D∞-TLI and MEB extracts width function which is less affected by unreasonable artificial fluctuation and has a *MARE* lower than 3%.

1. Introduction

As an important feature of overland flow, flow distance is important in many fields such as hydrology, geomorphology, and ecology. It plays a vital role in runoff or flood analysis (Bogaart and Troch, 2006; Di Lazzaro et al., 2016; Liu et al., 2012; McGuire et al., 2005; Xu et al., 2018), soil erosion or thickness simulation (Dong et al., 2022; Hickey, 2000; Tesfa et al., 2009), and water quality modeling (Fan et al., 2015). These researches rely on flow distance estimations at different scales, including the distance to channel or outlet (Bogaart and Troch, 2006; Van Nieuwenhuizen et al., 2021), the river length (Fan et al., 2015) and the uphill slope line length (Dong et al., 2022; Tesfa et al., 2009). Additionally, the width function of a hillslope or catchment, which is a representation of flow distance distribution, is commonly utilized as a hydrologic response function in hydrologic modeling (Bogaart and Troch, 2006; Gupta et al., 1986; Hazenberg et al., 2015; Lapides et al., 2022; Li et al., 2011; Liu et al., 2016; Moussa, 2008; Mutzner et al., 2016; Noël et al., 2014; Ranjram and Craig, 2021; Rigon et al., 2016; Troch et al., 2003). However, existing algorithms for the estimation of flow distance and width function have some noticeable limitations as outlined below, so there has been renewed interest in developing new algorithms to accurately estimate these two features.

Due to the discretized representation of terrains using grid digital elevation models (DEMs), flow distance is typically estimated by cumulating the length along the predicted DEM-based flow path (Mayorga et al., 2005). With the estimated flow distance distribution, the width function can be implicitly estimated (e.g., Liu et al., 2012; Sahoo and Sahoo, 2019b). This implicit method defines the width function as an area distribution function or a probability density function of equidistant belts (Bogaart and Troch, 2006; Liu et al., 2012; Moussa, 2008; Sahoo and Sahoo, 2019a, 2019b; Veneziano et al., 2000). There are also other simplified approaches to obtain the width function, such as taking the length of contour line as width (Fan and Bras, 1998) or generalizing the terrains into regular shapes for monotonically varying width function (Lapides et al., 2022; Noël et al., 2014; Ranjram and Craig, 2021). But the implicit method can provide more details for the applications in the real-world hillslopes or catchments than these simplified methods (Sahoo and Sahoo, 2019b).

Unlike the small number of width function estimation methods, there are more choices available for flow distance calculation. Each flow distance calculation method is a combination of one flow path extraction strategy and one cumulative length computation strategy. Over the past few decades, many algorithms were proposed to extract flow path (e.g., Chantaveeroda et al., 2023; O'Callaghan and Mark, 1984; Orlandini

* Corresponding author. The National Key Laboratory of Water Disaster Prevention, Hohai University, Nanjing, 210024, China.
E-mail addresses: wpf@hhu.edu.cn (P. Wu), jtliu@hhu.edu.cn (J. Liu), pungmiyom@gmail.com (M. Feng), lhayz@lzb.ac.cn (H. Liu).

https://doi.org/10.1016/j.envsoft.2023.105865
Received 21 June 2023; Received in revised form 5 October 2023; Accepted 26 October 2023
Available online 28 October 2023

et al., 2003; Quinn et al., 1991; Shin and Paik, 2017; Tarboton, 1997; Wu et al., 2020, 2022). Based on whether a cell is permitted to drain to more than one cell, these algorithms can be categorized into two types: the single flow direction (SFD) algorithms and the multiple flow direction (MFD) algorithms (Wilson et al., 2007). Due to the discretized grid structure of DEM, both SFD and MFD typically provide zigzag lines connecting neighboring cell centers. The traditional and simplest cumulative length algorithm adopts SFD and directly measures the length of the path as flow distance (Blodgett et al., 2021; Kotyra and Chabudziński, 2023; Li et al., 2020). Meanwhile, for the dispersive network generated by MFD, its flow distance is the weighted average length of all the lines (Bogaart and Troch, 2006).

However, the zigzag predicted flow path is inconsistent with the true flow path because the latter is smoother and may not pass the downslope cell centers (Paik, 2008). Consequently, the length of a predicted flow path is always longer than the true flow distance (Paz et al., 2008; Liu et al., 2012). One solution to this problem is to utilize some highly accurate algorithms to track the gravity-driven flow path, which is not constrained to pass the downstream cell centers (e.g., Zhou et al., 2011). But this solution is not feasible for large scales because of its high time and space consumption (Zhou et al., 2011), which must be taken into account in any practical application (Farahbakhsh et al., 2020; Masdari et al., 2020).

To ensure acceptable computational efficiency, the process of flow distance assignment for each cell center is better to only search in a local window rather than the whole flow path. Bogaart and Troch (2006) proposed such a two-segmented-distance framework that models the sub-distance from the cell center to a crossing point at a 3 × 3 window boundary and adds it to the sub-distance originating from the crossing point to a given target (e.g., channel) for the final flow distance value. But this method previously only adopted center-to-center flow path because the true crossing points inconsistent with cell centers may not be assigned flow distance values.

Some strategies have been introduced to mitigate this error. For instance, Paz et al. (2008) applied the fixed distance transform value of 0.96194 as an averaged approximation to shorten the computed flow distance, following the suggestions of Butt and Maragos (1998) as well as De Smith (2004). Liu et al. (2012) used the cosine of the angle between the SFD direction and the steepest downstream direction as the transform value to represent the specific bias in every 3 × 3 window. Dong et al. (2022) introduced a merging method to correct the flow distance, which used the Euclidean distance of every three local flow segments to replace the cumulative length. But all these methods aim to restore the true flow distance within a local window, and the absence of flow distance values at the window sides appears to still restrict the improvements in flow distance estimation.

Meanwhile, the missing flow distance values prevent it from dividing a cell covering multiple equidistant belts into correct numbers of belts. The conventional method adds the whole cell into merely one equidistant belt based on the flow distance value assigned for the cell center. Then the imprecise equidistant belt area function will result in a width function with unreasonable artificial fluctuation because some equidistant belts may encompass excessive regions from adjacent belts as illustrated in Fig. 1 and the results of some existing studies (e.g., Moussa, 2008; Liu et al., 2012; Sahoo and Sahoo, 2019b; Veneziano et al., 2000).

Overall, the absence of flow distance values in a cell region seems to affect the accuracy of flow distance estimation as well as width function estimation, thereby limit the development of new algorithms. Following some other geographical studies, the issue of missing values can be solved by any interpolation method (e.g., Schwendel et al., 2012; Yilmaz, 2007; Zanella-Béguelin et al., 2023). However, no existing study in this research domain attempting to bring in the interpolation method. Therefore, the potential rationality of employing the interpolation algorithm to improve the accuracy of flow distance estimation as well as width function estimation motivates the authors to conduct this study. Here the triangulation with linear interpolation (TLI) method is selected because of its acceptable precision and ability to generate straight isolines in a cell region, allowing a cell to be divided into regular equidistant belts whose area can be measured explicitly for the width function (Sloan, 1987; Yilmaz, 2007; Zhang et al., 2016).

In this study, a two-segment-distance strategy modified from the method proposed by Bogaart and Troch (2006) is introduced to estimate the flow distance distribution over the DEM. Subsequently, a novel strategy is presented for more precise width function estimation. Both of these strategies rely on the TLI method. Finally, the proposed strategies are assessed against several existing methods. Both the numerical and the real-world terrains of different resolutions are adopted for the assessments.

2. Methodology and experiments

2.1. Flow distance and width function estimation

The proposed algorithm is described by Algorithm 1, where the flow distance estimation (Line 1–21) and the width function calculation (Line 22–29) can be viewed as two independent steps. The flow distance

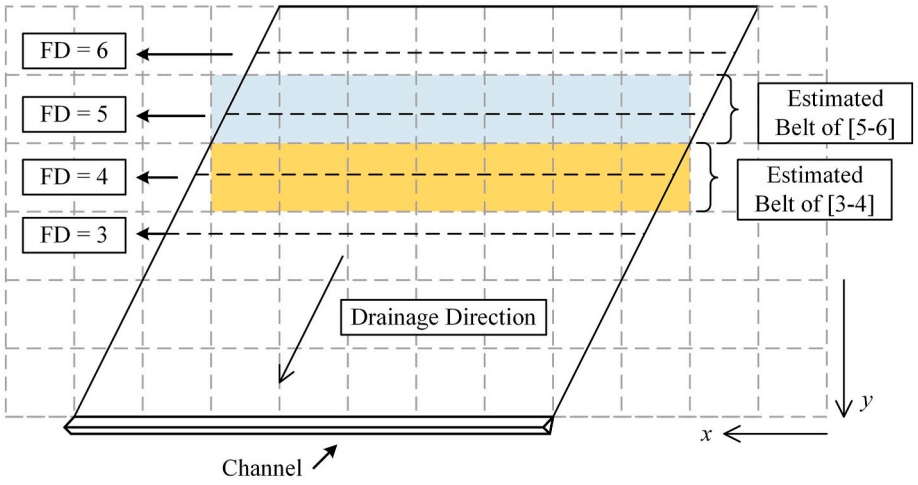


Fig. 1. The shortcoming of traditional width function algorithm. The 1 m-resolution DEM represents a plane whose radio between the gradients of x and y direction is 1:2. A cell may be divided into two parts by an equidistant line (black dashed line), but this cell can be added into only one equidistant belt based on the flow distance value assigned to its center. In the most serious cases, the flow distance value assigned to the center of any yellow or blue cell is 3.91 m and 5.03 m, respectively. So when the traditional method estimates the equidistant belt, the belt whose flow distance ranging from 4 m to 5 m is ignored.

estimation step necessitates a rasterized channel network layer and a DEM with flats and depressions removed. The width function calculation step demands a rasterized domain of the target hillslope or catchment. Detailed explanations of these two steps are provided below.

2.1.1. Flow distance estimation combining D∞ and TLI

The algorithm scans all cells located at the channel network or any other given target. The center of each selected cell is assigned a flow distance value of zero, and all the neighboring cells located at other positions are inserted into a min-first priority queue. The min-first priority queue can order the cells in ascending elevation automatically. Then, the algorithm continuously takes out the first cell of the priority queue to estimate the flow distance from its cell center (P_0) to channel (or other target) following the sketch in Fig. 2a.

As shown in Fig. 2a, a 3 × 3 window is firstly built using the current cell and its eight neighbors. The D∞ method proposed by Tarboton (1997) is adopted to provide the local flow path in the window. Then the crossing point (i.e., R) along the local flow path to the window boundary

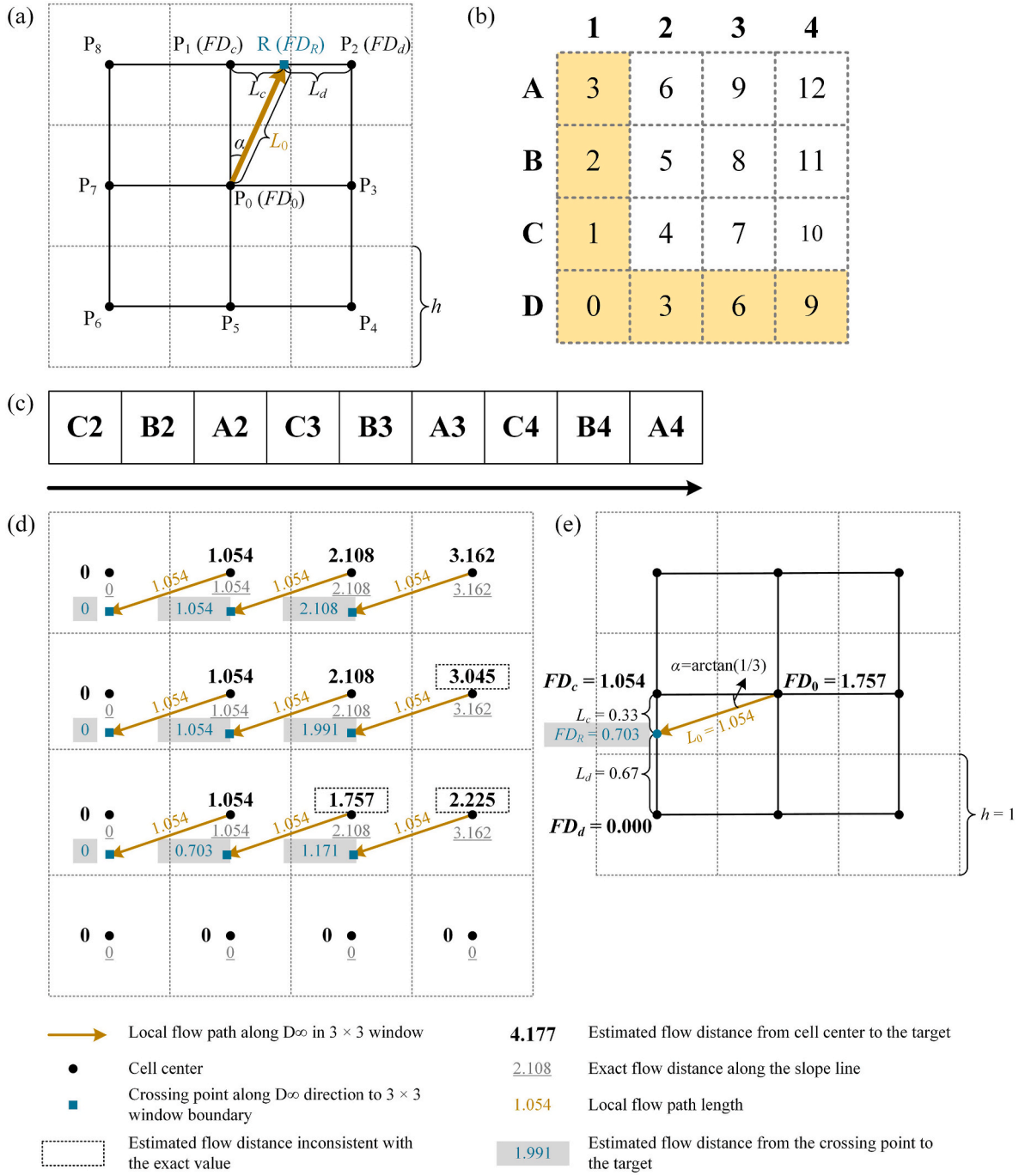


Fig. 2. The theory of the new flow distance algorithm. (a) Flow distance (FD_0) from a cell center (P_0) to a downstream target is computed as the sum of the local flow path length (L_0) along D∞ direction in a 3 × 3 window and the estimated flow distance from the crossing point (R) on the window boundary to the same target. (b) A 1 m-resolution DEM is adopted as an example, while the yellow cells are set to be channel cells. The cells out of the displayed region are set to another basin and hided for a clear display. The cells not belong to channels are processed following the order in (c). (d) The estimated distribution of flow distance from the cell centers to channel, as well as the data generated in the computational processes, are shown. (e) The enlarged window shows the process of the flow distance value assignment for C3.

can be identified. According to the D∞ theory, the cell centers of the closest cardinal neighbor and the closest diagonal neighbor (i.e., P_1 and P_2 in Fig. 2a) to R are always lower than P_0, so they have been assigned with flow distance values according to the framework iterating from low to high.

Although TLI cannot provide the entire flow distance distribution in the window before the assignment of flow distance for P_0, it can assign the flow distance value for any point on the line P_1P_2 with the existing flow distance values assigned to these two cell centers. The flow distance (FD_R) from R to the channel is computed by TLI as following:

$$FD_R = \frac{L_d FD_c + L_c FD_d}{L_c + L_d} \tag{1}$$

where FD_c and FD_d denote the flow distance from the nearest cardinal cell center (i.e., P_1) and the nearest diagonal cell center (i.e., P_2) to the channel, respectively. L_c and L_d denote the length from R to the nearest cardinal cell center (i.e., P_1) and the nearest diagonal cell center (i.e., P_2), respectively. L_c and L_d can be calculated as following:

$$L_c = h \tan \alpha \tag{2}$$

$$L_d = h(1 - \tan \alpha) \tag{3}$$

where h denotes the resolution of DEM, and α denotes the angle between the D∞ direction and the closest cardinal direction. There is a special case when the D∞ direction points to a neighboring cell center. In this case, the other nearest cell center may be higher than P_0, and its flow distance value may be unknown. It makes no difference because FD_R is equal to the flow distance value assigned for the cell center pointed by the D∞ direction according to Eq. (1) in this case. Then the local flow path length (L_0) by D∞ is calculated as:

$$L_0 = \frac{h}{\cos \alpha} \tag{4}$$

Finally, the flow distance (FD_0) from P_0 to channel is defined as:

$$FD_0 = FD_R + L_0 \tag{5}$$

After FD_0 is computed, the neighboring cells of the current cell is checked. Any neighboring cell that has not been assigned with values and is not in the priority queue will be inserted into the priority queue.

This new algorithm combines D∞ and TLI, so it is referred as D∞-TLI. The framework of D∞-TLI (Line 1–21 in Algorithm 1) adopts the priority queue to improve computing efficiency. This framework is based on the Priority-Flood method generalized by Barnes et al. (2014) and has a time complexity of $O(n\log_2 n)$. Here n is the number of cells in the DEM.

A DEM is provided as an example in Fig. 2b to demonstrate the capacity of D∞-TLI to restore the flow distance to channel. According to the order with elevation from low to high, the cells not belonging to channels are processed following the sequence in Fig. 2c. Then the estimated local flow path length, as well as both the estimated flow distances from the crossing point and the cell center to the DEM side, is labelled for each cell in Fig. 2d. Moreover, an enlarged 3 × 3 window in Fig. 1e is used to show the detailed assignment of flow distance for a cell center following Eqs. (1)–(5). More details about how the cells are processed one by one are provided in Fig. S1.

According to the estimated flow distance distribution (black bolded values) in Fig. 1d, the difference between the estimated flow distance and the exact slope line length is little for most cell centers. Here the exact slope line originating from a cell center is consistent with the gravity-driven flow path (Maxwell, 1870; Orlandini et al., 2014). This example only shows the potential of D∞-TLI to provide the reasonable flow distance distribution. However, whether more bias may appear when D∞-TLI is applied to other terrains (e.g., divergent, or convergent terrains) still needs further verifications. Hence, multiple terrains (see in Section 2.2.1) with different complexities are adopted to assess D∞-TLI in our experiments.

2.1.2. Width function calculation with TLI

The width function coincides with the area distribution function of equidistant belt (Moussa, 2008), so the proposed algorithm extracts the equidistant belt firstly. In this step, cells in the specific hillslope or catchment are processed one by one. For each cell, the proposed algorithm firstly obtains the equidistant lines whose flow distances to channel are multiple to the belt interval, then each area between two equidistant lines is added to the corresponding equidistant belt. Here the belt interval is always set to be the DEM resolution (e.g., Liu et al., 2012; Moussa, 2008; Sahoo and Sahoo, 2019b).

To implement above scheme, a given cell is firstly divided into eight triangular facets as marked in blue in Fig. 3a. Then every facet is processed independently. The grey facet in Fig. 3a is taken for illustration. There are two vertexes (i.e., V_c and V_d) except the current cell center (i.e., P_0) within this facet. V_c is the midpoint of the given cell center (P_0) and its cardinal neighbor center (P_1 for the selected facet), and V_d is the midpoint of the given cell center (P_0) and its diagonal neighbor center (P_2 for the selected facet).

Equidistant lines (dotted lines in Fig. 3b) should be computed firstly. However, if any one of the neighboring cells (i.e., P_1 or P_2) belong to another hillslope or catchment, this facet will not be further divided to avoid the effects of unreasonable critical lines. Then this facet is added to the belt covering the flow distance from the cell center (P_0). This step ensures that the integration of the width function is equal to the area of the extracted hillslope or catchment.

When all the three vertexes belong to the same hillslope or catchment, the continuous flow distance field for the facet is calculated using the TLI method. Here the flow distance from P_0, P_1, P_2, P_3 are expressed as FD_0, FD_1, FD_2, FD_3, and the mean of these four flow distance values is expressed as MD. According to the TLI, the flow distance from V_c to channel (i.e., FD_{vc}) can be calculated following:

$$FD_{vc} = (FD_0 + FD_1)/2 \tag{6}$$

According to Zhou et al. (2011) method, the flow distance from V_d to channel (i.e., FD_{vd}) is calculated as below:

$$FD_{vd} = \begin{cases} (FD_1 + FD_3)/2, if\ |MD - (FD_1 + FD_3)/2| \le |MD - (FD_0 + FD_2)/2| \\ (FD_0 + FD_2)/2, if\ |MD - (FD_1 + FD_3)/2| > |MD - (FD_0 + FD_2)/2| \end{cases} \tag{7}$$

Then the equations for the flow distance from any point at the facet sides to channel are shown in Fig. 3b. These equations can also help to obtain the point position at a side with a given flow distance. Thus, when the given flow distance ranges between the minimum and the maximum flow distances from the three vertexes, only two points owning the given distance can be found at the three sides, and the equidistant line can be approximated by a straight line linking them. Thereupon, the facet can be divided into multiple equidistant belts between the equidistant lines (Fig. 3c and d).

After all the cells in the hillslope or catchment are processed, the probability density ($p(x)$) and the area distribution of equidistant belt are generated (Fig. 3e). Here the width of every equidistant belt is the ratio of the equidistant belt area ($Sp(x)$) to the belt interval (L_u), i.e., $Sp(x)/L_u$, and is defined as the width at the middle flow distance of the belt, where S is the total area of the hillslope or catchment (Fig. 3f). Hence, the applications of area distribution and probability density function to obtain the width function are consistent.

This proposed division method for width function is referred as the multiple equidistant belt (MEB) method, which is different from the conventional method adds a cell into single equidistant belt (SEB). The implementation of MEB requires no iteration or other complex processes, and its time complexity is $O(n)$.

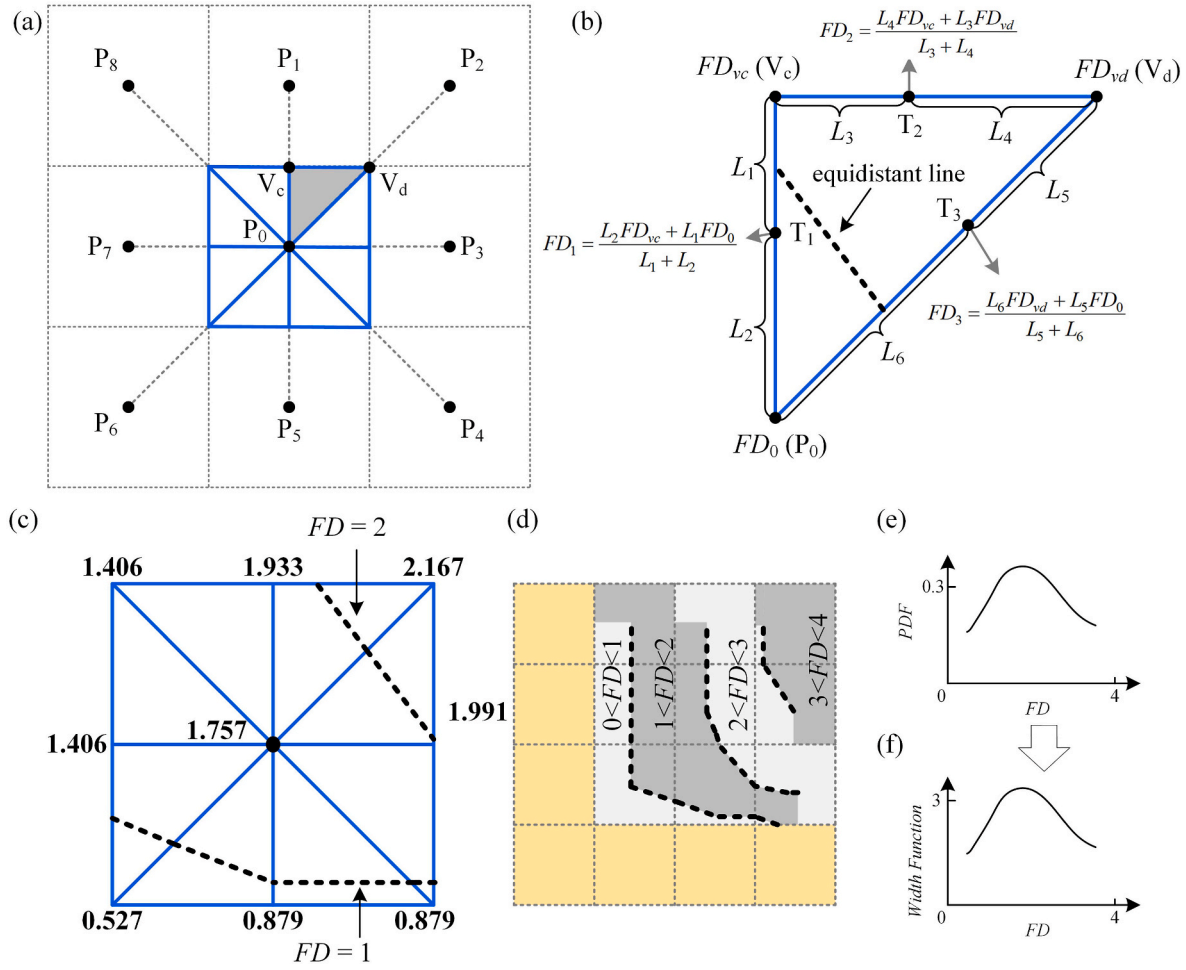


Fig. 3. The method to determine the local equidistant belt area which is used to constitute the width function. (a) The cell is divided into eight facets as shown with blue boundary, and the flow distance of any point at the facet boundary can be calculated following (b), where L_1, …, L_6 denote the lengths from the points to the vertexes. Each equidistant line is straight and linking two points at the boundary with the same flow distance. (c) The cell C3 in Fig.1b is used as an example to shown the equidistant line, and the equidistant belt distribution for the region out of channels in Fig.1b is shown in (d). It is notable that the equidistant lines cannot reach the region sides. Then (e) the PDF of the equidistant belt area and (f) the width function can be calculated.

2.2. Experiment materials and assessment criteria

2.2.1. Numerical and real-world terrains

Four numerical terrains and two real-world terrains are adopted for algorithm assessments (Fig. 4). The numerical terrains contain an ellipsoid, an inverse ellipsoid, a plane, and a saddle (Fig. 4a–d). These terrains represent divergent, convergent, plain and complex terrains, respectively. The formulas proposed by Li et al. (2021) are used to build these terrains with six resolutions (1 m, 2 m, 5 m, 10 m, 20 m, 30 m). As mentioned in Section 2.1, the exact flow distance for the numerical terrains is equivalent to the slope line length. This length can be calculated by integration using the slope line formulas introduced by Li et al. (2021), then the exact equidistant belt area can be divided for the exact width function. When assessing the flow distance algorithms, only the partial ellipsoid with a square boundary in Fig. 4i is used because there is a systematic error between the circle boundary of the complete ellipsoid and the valid DEM cells (Fig. S2). But the assessments of the width function use the complete ellipsoid because it is hard to obtain the exact equidistant line or belt with integration when only the partial terrain is used. The saddle is not employed for the width function extraction due to the lack of the exact equidistant belt area.

It is difficult to obtain the exact flow distances from most positions over a real-world terrain to the channel or outlet. However, flow distance from a point in a channel or gully can be measured along the overland flow trajectory. Therefore, some channels (or gullies) in two real-world terrains, including a sub-basin of the Spruce Canyon (Fig. 4e) and the Duodigou Basin (Fig. 4f), are mapped using the images from the Google Earth for the assessments (e.g., Fig. 4g and h).

The Spruce Canyon is in New Mexico, USA, and the selected tributary sub-basin (referred as SCT Basin) owns a drainage area of 7.0 km^2 with the elevation ranging from 2292 m to 3027 m. This basin has a relatively low mean slope (17.2°), while the valley bottoms and the channels are narrower than 100 m and 3 m, respectively. The downstream channels of thirty points are mapped. To avoid repetitive computation bias, only two longest channels are selected as the main channels, while other twenty-eight channels end up at the main channel and are regarded as branches. The lengths of two main channels are 3475 m and 3496 m, while the lengths of the branches range from 75 m to 687 m. Bare earth DEM data with 1 m resolution is provided by the Jemez River Basin Snow-off LiDAR Survey, and is resampled to five coarser resolutions (2 m, 5 m, 10 m, 20 m, 30 m) consistent with the selected resolutions of the numerical terrains.

The Duodigou Basin (referred as DDG Basin) is located in the Tibetan Plateau, China (Fei et al., 2022). It is a steep alpine terrain covering 56.6 km^2 with the elevation ranging from 3719 m to 5425 m and a mean slope equal to 28.0°. The valley bottom is narrow (<20 m) at the upstream and wide (>900 m) at the downstream. The downstream channels of ten points are mapped. The longest channel (11386 m) is selected

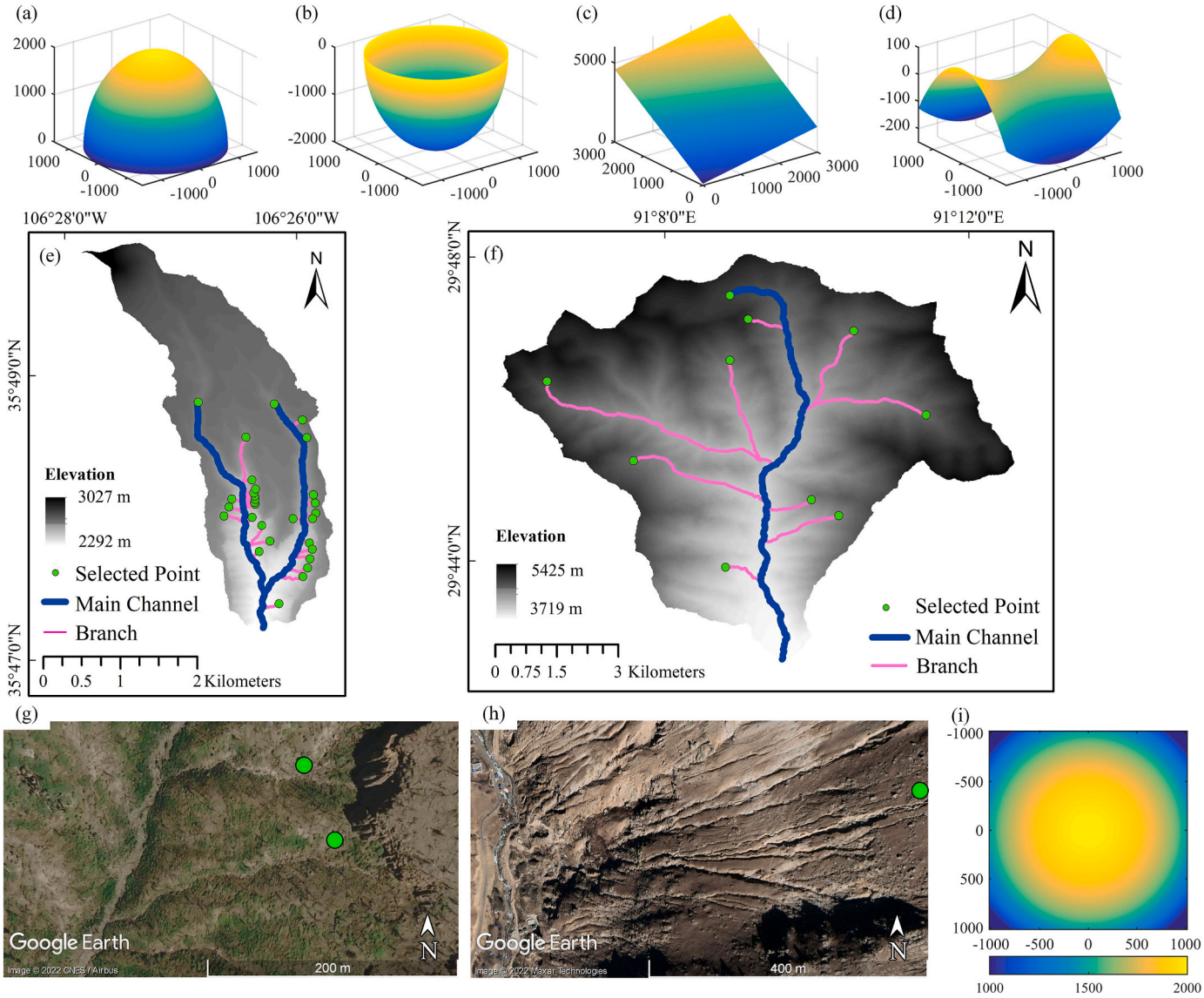


Fig. 4. Four numerical and two real-world terrains are used for the assessments, including (a) an ellipsoid, (b) an inverse ellipsoid, (c) a plane, (d) a saddle, (e) the SCT Basin, and (f) the DDG Basin. Local images of (g) the SCT Basin and (h) the Duodigou Basin with several selected source points mapped are used to show the branch channels or gullies. Moreover, the elevation distribution of the square partial region of the ellipsoid (-1020 m $< x$, y < 1020 m) used for the flow distance assessments is shown in (i).

Table 1
The flow distance algorithms adopted for the comparison in this study.

Algorithm	Origin of the cumulative distance algorithm	Adopted flow direction algorithm	Origin of the flow direction algorithm
D8-CL	–	D8	O'Callaghan and Mark (1984)
D8-DT	Paz et al. (2008)	D8	O'Callaghan and Mark (1984)
D8-CT	Liu et al. (2012)	D8/D∞	O'Callaghan and Mark (1984)/Tarboton (1997)
iFAD8-CL	–	iFAD8	Wu et al. (2020)
iFAD8-DT	Paz et al. (2008)	iFAD8	Wu et al. (2020)
iFAD8-CT	Liu et al. (2012)	iFAD8/D∞	Wu et al. (2020)/ Tarboton (1997)
iFAD8-M	Dong et al. (2022)	iFAD8	Wu et al. (2020)
QMFD-CL	Bogaart and Troch (2006)	QMFD	Quinn et al. (1991)
D∞-CL	Bogaart and Troch (2006)	D∞	Tarboton (1997)
D∞-TLI	Current study	D∞	Tarboton (1997)

as the main channel. The lengths of the branches range from 936 m to 5979 m. These channels are narrower than 10 m. The 12.5 m-resolution DEM of this basin is obtained from the Advanced Land Observing Satellite (ALOS). Limited by the coarse initial resolution, the DEM is resampled to four resolutions (15 m, 20 m, 25 m, 30 m) with smaller intervals than those of other terrains to show the influence of the resolution.

2.2.2. Algorithm assessments

The estimated flow distance distributions estimated by ten flow distance algorithms, including D∞-TLI and nine other algorithms, are adopted by the assessments over the numerical or the real-world terrains. The information of these algorithms are listed in Table 1. Here the traditional cumulative length (CL) method, the distance transform (DT) method by Paz et al. (2008), and the cosine transform (CT) method by Liu et al. (2012) are combined with two SFD algorithms, namely the classical D8 algorithm (O'Callaghan and Mark, 1984) and a highly precise algorithm named iFAD8 (Wu et al., 2020). The merging (M) method by Dong et al. (2022) is also employed and combined with iFAD8. The cumulative length methods based on D∞ and QMFD proposed by Bogaart and Troch (2006) are also adopted. Hence, ten algorithms for assessments are D8-CL, D8-DT, D8-CT, iFAD8-CL, iFAD8-DT, iFAD8-CT, iFAD8-M, D∞-CL, QMFD-CL and D∞-TLI. The flow direction algorithms (D8, iFAD8, D∞ and QMFD) are selected due to their applicability for flow distance measurements. Although some other flow direction algorithms are shown to be more effective in other applications (e.g., Pilesjö and Hasan, 2014; Wu et al., 2022), they are not suitable to this study because they provide flow path out from a non-point source.

The mean absolute relative error (*MARE*) is used to assess the deviations between the estimated and the exact flow distances. It is defined as follows:

$$RE_i = \frac{PV_i - EV_i}{EV_i} \tag{8}$$

$$MARE = \frac{1}{n}\sum_{i=1}^{n} |RE_i| \tag{9}$$

where PV_i and EV_i are the estimated and the exact values of the ith cell, respectively. RE_i denotes the relative error of the ith cell, and n denotes the number of cells considered for the assessment.

When width function is estimated for a terrain, the exact widths of some estimated equidistant belts may be zero due to the possible overestimation of the flow distance. This phenomenon can limit the direct application of *MARE* to assess the deviations between the estimated and the exact width functions. Hence, two valid assessment criteria are adopted here. Firstly, a part of the equidistant belts with flow distances not exceeding the maximum exact flow distance are selected to calculate the *MARE* following Eqs. (8) and (9). Here PV_i and EV_i in Eq. (8) are the ith estimated and the ith exact widths, respectively. Then the exceeding index (*EI*) is adopted to represent the proportion of the widths whose estimated flow distances are longer than the exact maximum flow distance.

$$EI = \frac{\sum_{j=1}^{m} W_j}{T} \tag{10}$$

where m denotes the number of the equidistant belts exceeding the maximum exact flow distance, and W_j denotes the jth exceeding equidistant belt width. T denotes the total width of all exact equidistant belts. It is obvious that an accurate width function should possess both low *MARE* and *EI* values.

3. Results

3.1. Assessments of the flow distance algorithms

3.1.1. Performances over the numerical terrains

To clearly illustrate the difference between the exact and the estimated flow distance distributions, the partial enlarged details over the 20 m-resolution terrains are shown in Fig. 5. Meanwhile, the flow distance distributions over the whole terrains are provided in Fig. S3. According to Fig. 5b and d, it is evident that D∞-TLI accurately reproduce the exact flow distance for the inverse ellipsoid and the saddle. The isolines of flow distance by D∞-TLI are smooth and parallel to the exact isolines over the partial ellipsoid, with a deviation as low as iFAD8-CT or iFAD8-M (Fig. 5a). However, D∞-TLI underestimates the flow distance where the exact isolines facing two directions intersect over the plane (see in Fig. 5c). In contrast, the results by the D8-based algorithms (including D8-CL, D8-DT and D8-CT) are unsatisfactory. D∞-CL and QMFD-CL overestimate flow distance over every numerical terrain. Although iFAD8-CL and iFAD8-DT lead to large deviations, the other two iFAD8-based algorithms, iFAD8-CT and iFAD8-M, are able to reproduce the exact flow distance distribution more reliably than other algorithms except D∞-TLI. But the accuracy of iFAD8-CT or iFAD8-M is unsteady as indicated by the undulant isolines, and it is obviously lower than the accuracy of D∞-TLI over the inverse ellipsoid and the saddle. Hence, the visual assessments demonstrate that D∞-TLI is the best choice to reproduce flow distance distributions over the numerical terrains, a conclusion that is further supported by the results over 5 m-resolution terrains (Fig. S4).

According to the quantitative assessments (Fig. 6), lower *MARE* appears when a finer resolution is used for most cases. D∞-TLI emerges as the superior algorithm because it obtains the least *MARE*s across all resolutions over the inverse ellipsoid, the plane or the saddle, while only iFAD8-CT has a comparable performance with D∞-TLI over the partial ellipsoid. Overall, the average *MARE* of D∞-TLI is a mere 2.31 % over four numerical terrains. iFAD8-CT and iFAD8-M are the two algorithms follow closely behind D∞-TLI, exhibiting obvious improvements to iFAD8-CL and iFAD8-DT while iFAD8-DT outperforms iFAD8-CL. When comparing the D8-based algorithms, D8-DT and D8-CT outperform D8-CL over the partial ellipsoid and the inverse ellipsoid, but underperform D8-CL over the saddle. The D8-based algorithms have similar performances over the plane. Consistent with the results in Fig. 5, D∞-CL and QMFD-CL produce great errors over all the terrains, but D∞-CL seems to be more accurate than QMFD-CL.

3.1.2. Real-world applications

The distribution of flow distance to the mapped channels is calculated in both the SCT Basin and the DDG Basin, and the visual results by

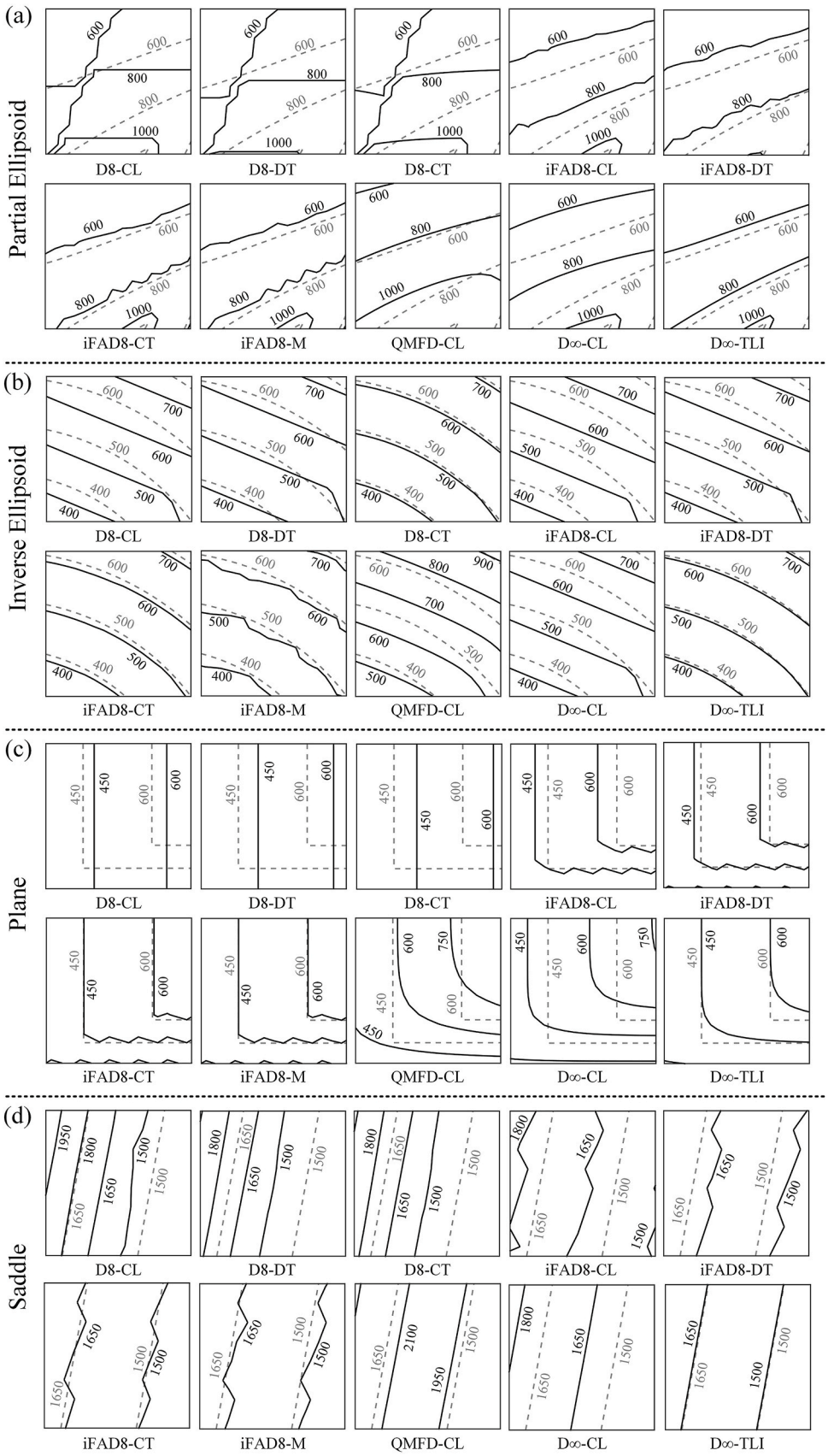


Fig. 5. Enlarged windows of the exact flow distance distributions (grey dashed lines) versus estimated flow distance distributions (black solid lines) to the terrain boundary by different algorithms over four numerical terrains with 20 m resolution. The window ranges of the partial ellipsoid or the inverse ellipsoid are 100 m < x < 400 m and 300 m < y < 600 m. The window ranges of the plane are 350 m < x < 650 m and 100 m < y < 400 m. The window ranges of the partial ellipsoid are 150 m < x < 300 m and 1250 m < y < 1400 m.

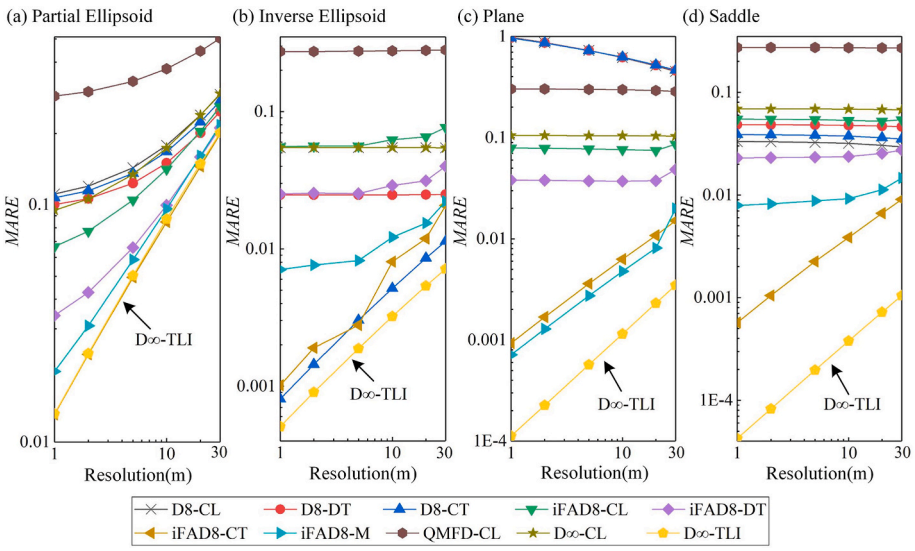


Fig. 6. The mean absolute relative error (MARE) of the estimated flow distance by different algorithms over (a) the partial ellipsoid, (b) the inverse ellipsoid, (c) the plane, and (d) the saddle with six different resolutions. The plots of D∞-TLI are specially marked. Here the lines of D∞-TLI and iFAD8-CT are almost coincident for (a) the partial ellipsoid.

different algorithms are shown in Fig. 7. Here some short channels in the SCT Basin are ignored and only four channels are adopted. Enlarged details in Fig. 7c and d reveal that the MFD-based algorithms, including D∞-TLI, D∞-CL and QMFD-CL, yield smoother isolines of flow distance than the seven selected SFD-based algorithms. In contrast to other algorithms, D∞-CL and QMFD-CL always overestimate the flow distance. The application of strategy DT, CT and M effectively shorten the results of CL no matter which SFD algorithm (D8 or iFAD8) is selected. This is evident from the isoline of 100 m over the SCT Basin (Fig. 7c) and the isoline of 600 m over the DDG Basin (Fig. 7d).

Two cases are taken into consideration for quantitative assessments: the flow distances from the selected points to the main channels shown in Fig. 4e and f, and the flow distances from all the selected points to the basin outlet. The wavy *MARE*s show that the ability of every selected algorithm is unsteady over the real-world terrains with different resolutions (Fig. 8). Nonetheless, compared with other algorithms, D∞-TLI consistently yields acceptable *MARE*s. For any given case in Fig. 8, the average *MARE* of all the resolutions is listed in Table 2. D∞-TLI ranks among the top three algorithms for any case. In addition, the average *MARE* of all the cases and resolutions are calculated, with D∞-TLI achieving the lowest value of 4.01 %. The values of D8-CT (4.30 %) and iFAD8-CT (4.29 %) are lower than other algorithms except for D∞-TLI. The average *MARE*s of QMFD-CL are excessively high over the real-world terrains. The performances of D8-CL, D8-DT, iFAD8-CL, iFAD8-M and D∞-CL are similar according to the results in Table 2.

3.2. Assessments of the width function

According to the results in Section 3.1, only five flow distance algorithms are selected to provide the flow distance distribution for the assessments of width function algorithms, including D∞-TLI, classical D8-CL, and three algorithms with acceptable performances over the numerical or the real-world terrains (i.e., D8-CT, iFAD8-CT and iFAD8-M). Fig. 8 shows the width functions estimated by two width function algorithms (i.e., the conventional SEB and the proposed MEB) with different estimated flow distance distributions over three 20 m-resolution numerical terrains. Here the flow distance interval of the equidistant belt for the width function is equal to the DEM resolution (i.e., 20 m) following some existing studies (e.g., Liu et al., 2012; Moussa, 2008; Sahoo and Sahoo, 2019b).

As shown in Fig. 9, while the exact width functions for the numerical terrains are smooth, SEB causes artificial fluctuations in the estimated width functions in most cases. The MEB algorithm effectively mitigates these unreasonable artificial fluctuations successfully. The quantitative assessment results in Fig. 10 further demonstrate that MEB improves the accuracy of the estimated width function, yielding lower *MARE* than SEB for any selected flow distance distribution. Meanwhile, *EI* is always equal to zero over the ellipsoid or the plane, and slightly larger than zero over the inverse ellipsoid when the flow distance distribution by D8-CL or iFAD8-M is adopted. The estimated width function combining MEB and D∞-TLI is highly consistent with the exact width function (Fig. 9m-o), and generally achieves the lowest *MARE* over all the numerical terrains (Fig. 10). The average *MARE* is 2.97 % for this combination but higher than 5% for other combinations. Other estimated flow distance distributions except the distribution by D∞-TLI can restore the trend of the exact width function over the ellipsoid with SEB or MEB (Fig. 9a, d, 9g and 9j). However, the deviation of D8-CL is great over the inverse ellipsoid (Fig. 9b) and the plane (Fig. 8c), while D8-CT exhibits substantial deviations over the plane (Fig. 9f). These unreasonable deviations can also be identified from Fig. 10.

The application of MEB over the real-world terrain also exhibits its ability to overcome the artificial fluctuations (Fig. 11). To obtain the exact width function over a real-world terrain is a great challenge. However, if the flow distance to channel is close to zero, the estimated width can be approximated as twice the channel length because the equidistant belt has a small interval and is proximity to both the channel banks. This is not a very disciplined assessment method, but can provide a reference for the application over the real-world terrain. Here the SCT Basin with 1 m-resolution DEM is selected for real-world applications and the width function to channel (including four channels in Fig. 7a) with an equidistant belt interval of 1 m is calculated. No matter which algorithm is adopted to determine the flow distance distribution, the area of the first equidistant belt with flow distance ranging from 0 m to 1 m should be much smaller than the exact area as shown in Fig. 9. This is because the exact first equidistant belt is covered by both the hillslope and the channel cells while the proposed algorithm only estimates width function using the hillslope cells. Hence, the next belt ranging from 1 m to 2 m (i.e., the width at 1.5 m flow distance) is used to predict the total channel length (7546 m), and the results are shown in Table 3.

The relative errors of all the predicted lengths by SEB exceed 10 %, while the relative errors by MEB are lower than 10 %. When combined with MEB, the relative errors of D∞-TLI and D8-CL are lower than 3 %.

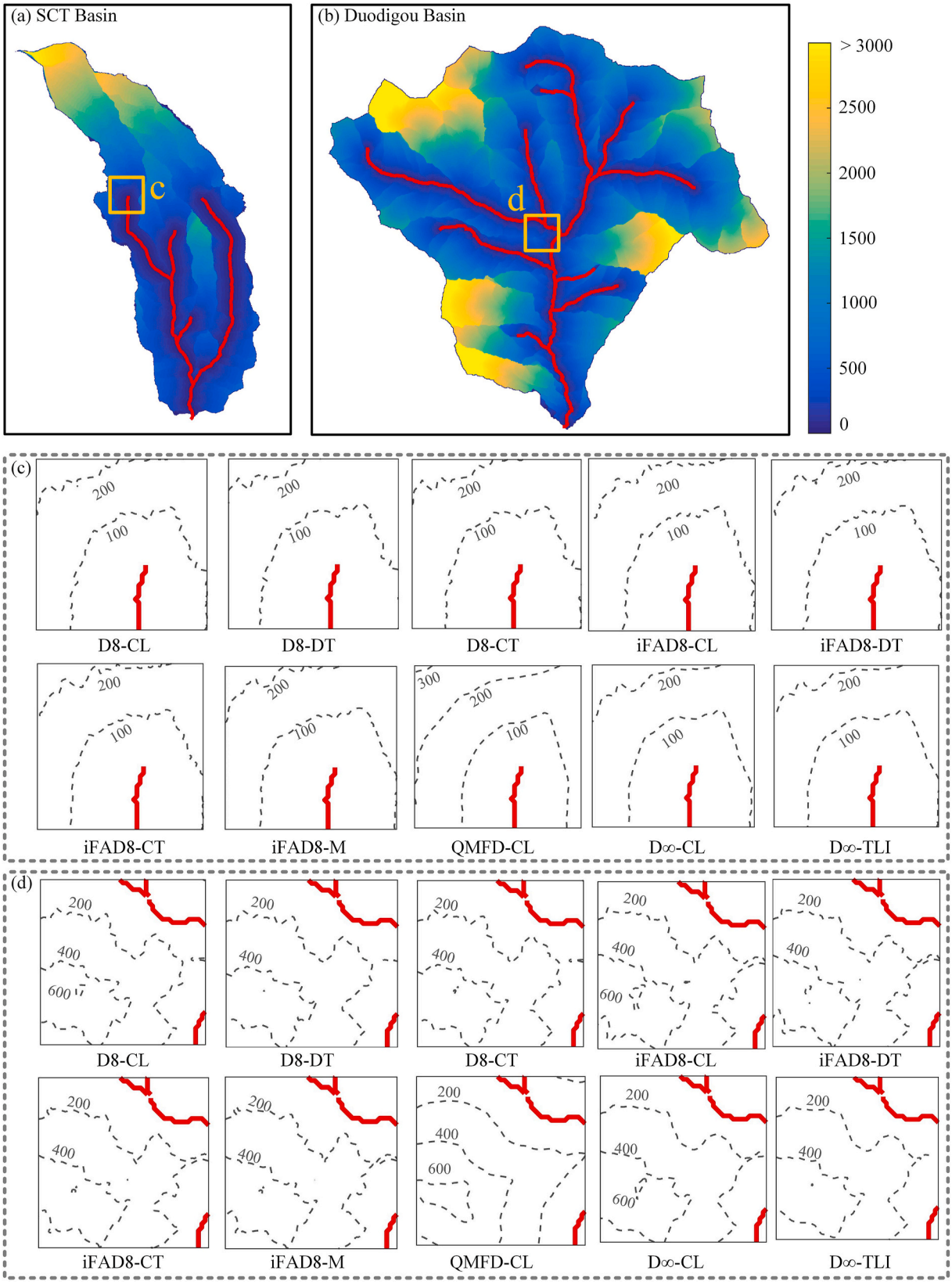


Fig. 7. The estimated flow distance distribution (dashed lines) to channels (solid lines) by D∞-TLI over (a) the SCT Basin and (b) the DDG Basin. For the marked domains in (a) and (b), enlarged windows in (c) and (d) are used to show the difference between the flow distance distributions estimated by ten selected algorithms.

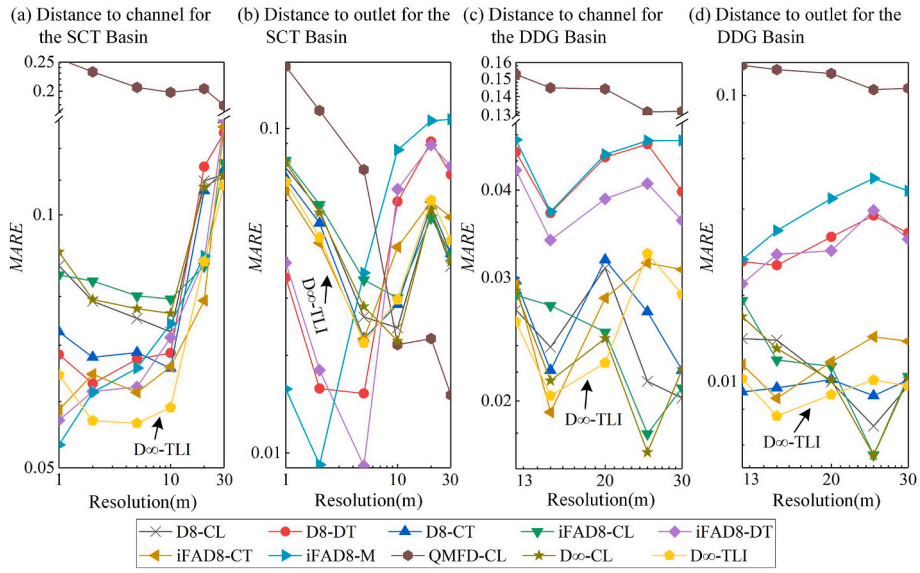


Fig. 8. The mean absolute relative error (MARE) of the estimated flow distance to channel or outlet for the selected points in the SCT Basin and the Duodigou Basin.

Table 2
The average MARE of the estimated flow distance by different algorithms.

	D8-CL	D8-DT	D8-CT	iFAD8-CL	iFAD8-DT	iFAD8-CT	iFAD8-M	QMFD-CL	D∞-CL	D∞-TLI
Distance to channel for the SCT Basin	8.93	8.44	8.24	8.84	**7.84**	**7.62**	7.92	21.26	9.04	**7.22**
Distance to outlet for the SCT Basin	**4.58**	4.81	**4.52**	4.94	4.93	4.78	5.99	6.71	4.65	**4.52**
Distance to channel for the DDG Basin	2.61	4.32	2.79	**2.48**	3.93	2.94	4.65	14.31	**2.52**	**2.57**
Distance to outlet for the DDG Basin	**1.09**	3.09	**0.96**	1.17	2.99	1.20	4.03	11.59	1.12	**0.93**
All the cases above	4.49	5.29	**4.30**	4.57	5.04	**4.29**	5.72	13.47	4.50	**4.01**

Note. The unit is in 10^{-2}, and the three lowest average *MARE*s of every case are bolded.

Table 3 presents the widths at three neighboring flow distances (i.e., 1.5 m, 2.5 m, 3.5 m) by different combinations. Although slight fluctuation in width function is normal, the fluctuation of the selected widths by any SEB-based combination is too strong for such a small flow distance interval. This unreasonable fluctuation is obviously artificial which is caused by SEB.

4. Discussions

4.1. Different algorithms on flow distance estimation accuracy

The flow direction algorithm selected appears to be the primary factor influencing the accuracy according to the results in Section 3.1. All the D8-based algorithms (i.e., D8-CL, D8-DT and D8-CT) provide abnormal flow distance distributions over the partial ellipsoid, the plane and the saddle, but yield more effective distributions over the inverse ellipsoid (Fig. 5). This is because D8 can generate false flow paths that direct the flow to incorrect targets over the divergent and the plain terrains, but has an acceptable performance over the convergent terrain (Wu et al., 2022). Meanwhile, D8-CL and D8-CT can provide reasonable flow distances to outlet for the selected points over two selected real-world terrains (Table 2), as the majority of the flow path from a selected point to the outlet traverses a convergent valley where the D8 algorithm operates effectively.

iFAD8 can provide reasonable zigzag flow paths out from the cell center (Wu et al., 2020), thus the results of any iFAD8-based algorithm (i.e., iFAD8-CL, iFAD8-DT, iFAD8-CT and iFAD8-M) can roughly reflect the features of the exact flow distance distributions. D∞ and QMFD provide dispersive flow paths and may drain a part of the flow into the channel at some unusually distant locations, so D∞-CL and QMFD-CL always overestimate the flow distance (Figs. 5 and 7). D∞ is less dispersive than QMFD (Orlandini et al., 2012), so D∞-CL obtains better results than QMFD-CL over all the terrains. Generally, the SFD-based algorithms get better application results than the existing MFD-based algorithms (i.e., D∞-CL and QMFD-CL).

D∞-TLI outperforms any other selected algorithm in estimating the generally most accurate flow distance over both the numerical and the real-world terrains. Although the traditional D∞ method is treated as a MFD method, D∞-TLI neglects the dispersive global flow path of D∞ and only employs D∞ for the local drainage direction. Hence, D∞-TLI does not suffer the serious problem of dispersive flow path by the MFD methods and can provide more accurate flow distance.

There are some strategies to improve the precision of the local drainage direction by D∞ (e.g., Hooshyar et al., 2016; Wu et al., 2020). These new methods may obtain the potential to further improve the accuracy of the estimated flow distance. But when the new infinite direction (ND∞) method proposed by Wu et al. (2020) is adopted to replace the D∞ direction in D∞-TLI, no obvious improvement appears to the accuracy over the real-world terrains (Fig. S5). That is because the limited improvement of ND∞ to D∞ can be offset by other errors, such as the errors in TLI or the DEM generation. Hence, the applicability of the improved versions of D∞ requires more assessments in further studies, and the tradition D∞ direction is recommended in this study due to its simplicity and popularization.

4.2. Width function estimation

It is possible that some cells cover multiple equidistant belts when their cell centers are located on one equidistant belt (Fig. 1). Then SEB adds all these cell regions into one equidistant belt while other neighboring equidistant belts receive no cell area, which draws the artificial fluctuations in Fig. 9. The originality of MEB is to attempt to divide a cell

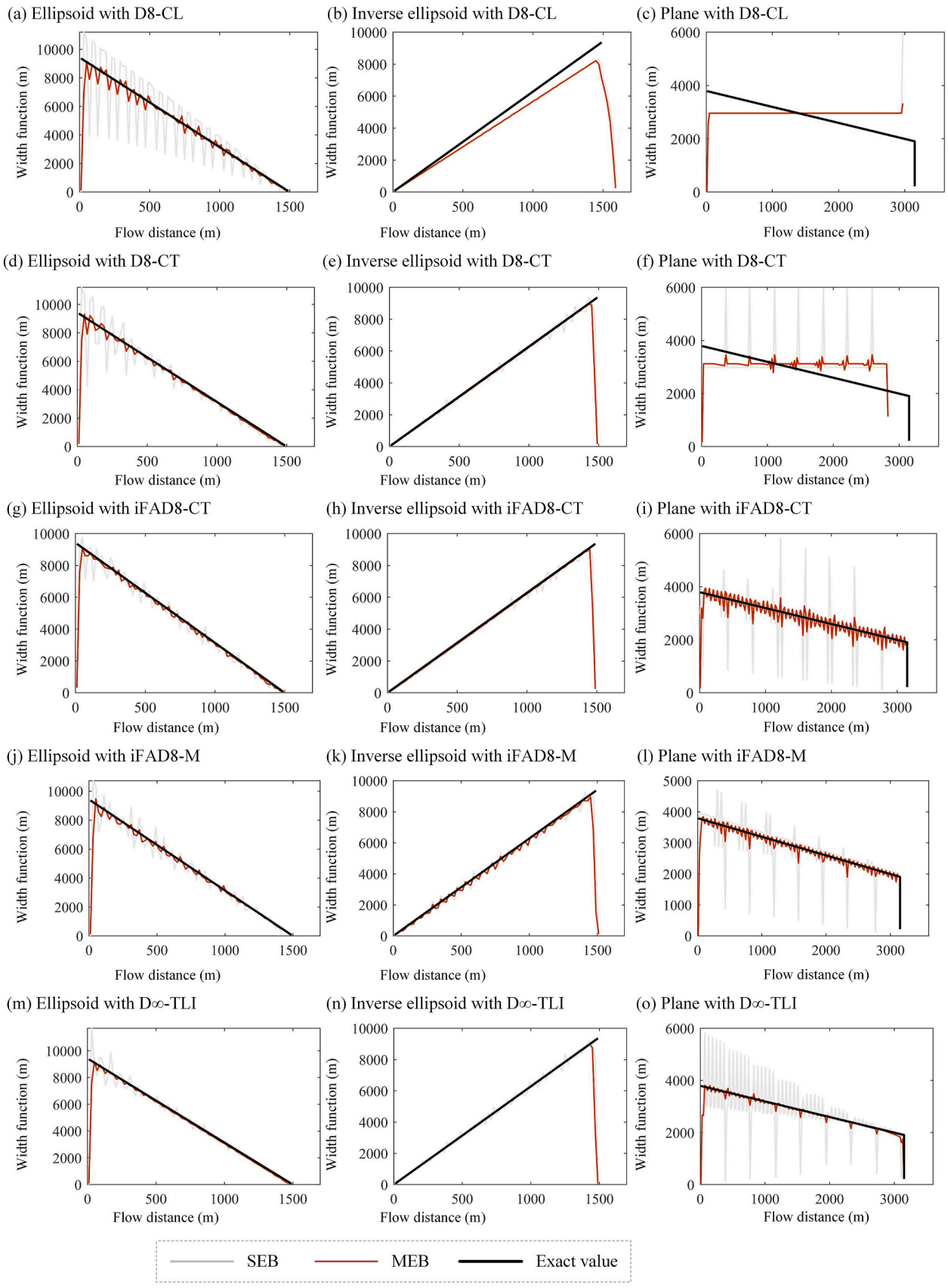


Fig. 9. The width functions derived by the SEB and the MEB method with five selected flow distance algorithms and 20 m-resolution numerical terrains. The flow distance interval is equal to the resolution.

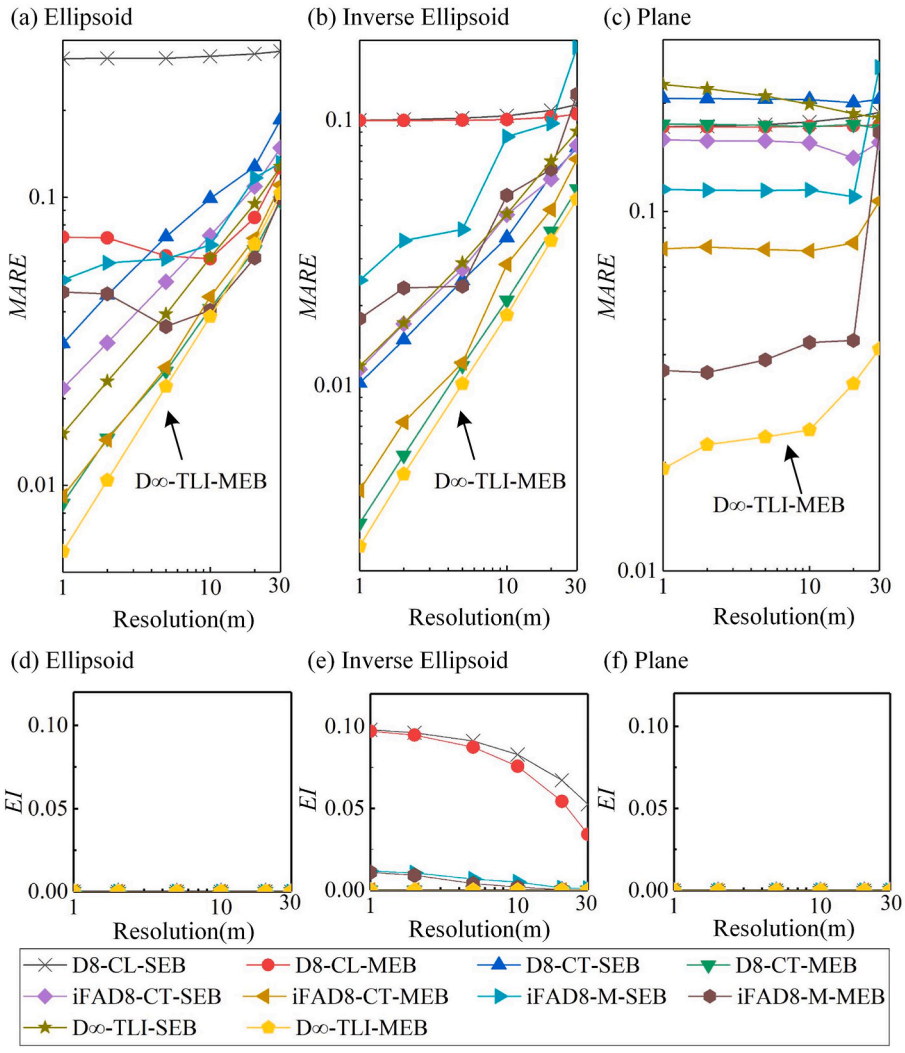


Fig. 10. The mean absolute relative error (MARE) and the exceeding index (EI) of the estimated width function over three numerical terrains with different resolutions. The flow distance interval is equal to the specific resolution.

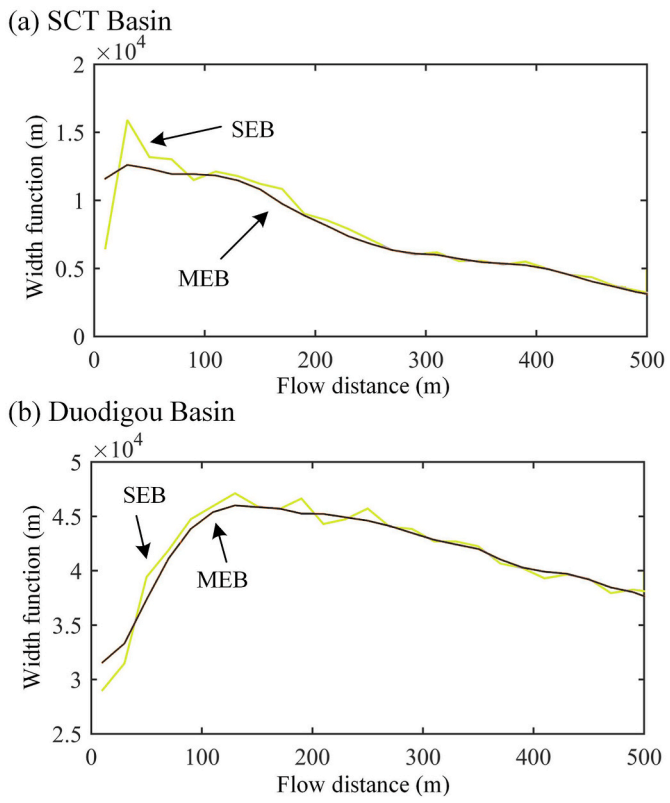


Fig. 11. The partial width functions with flow distance shorter than 500 m by the SEB and the MEB method. The distributions of flow distance to channel are estimated by D∞-TLI over (a) the SCT Basin and (b) the DDG Basin with the resolution of 20 m. The flow distance interval of the width function is equal to 20 m.

into correct equidistant belts, so it is unsurprising to find that MEB leads to slighter artificial fluctuation. However, the accuracy of the estimated width function depends not only on the width function algorithm (i.e., SEB or MEB), but also on the selected flow distance algorithm. With a specific flow distance distribution, MEB can estimated the width function more accurately than SEB (Fig. 9). For a specific width function algorithm (SEB or MEB), its accuracy can be greatly improved by using a more accurate flow distance algorithm. However, the combination of SEB with a more accurate flow distance algorithm may outperform the combination of MEB with a flow distance algorithm of which the accuracy is not that high. As an instance, iFAD8-CT causes higher *MARE* than iFAD8-M when providing flow distance distribution for the plane (Fig. 5c), but iFAD8-CT-MEB provides a better width function than iFAD8-M-SEB (Fig. 9c). While D∞-TLI is shown to be the best choice for flow distance estimation, the combination of D∞-TLI with MEB is optimal, which is demonstrated by the steady great performance over the numerical and the real-world terrains (Fig. 10 and Table 3).

The results in Section 3.2 are based on a traditional precondition that the equidistant belt interval for the width function is equal to the DEM resolution. Since this is a small interval, a large proportion of the whole cell region may fall outside the correct equidistant belt. This precondition may increase the artificial fluctuation. If the interval is set to be larger, more area in a cell can belong to the same equidistant belt as the cell center, then the SEB method will suffer a slighter artificial fluctuation. However, MEB can still optimize the accuracy of SEB to a degree in this case as shown in Fig. S6.

4.3. Computational efficiency

All the flow distance algorithms used in this study are implemented following the two-segmented-distance strategy. The runtimes for the plane with different resolutions show that this strategy can guarantee acceptable computational efficiency (Table 4). Although the runtime increases exponentially with the number of cells, D∞-TLI can process a DEM with more than 9×10^6 cells in less than 20.0 s. This runtime is similar to D∞-CL, and is longer than the D8-based algorithms as well as QMFD-CL, while is shorter than the iFAD8-based algorithms. This is similar to the difference in efficiency of the selected flow direction algorithms (D8, iFAD8, D∞ and QMFD) as shown by Wu et al. (2022). So, the selected flow direction algorithm seems to be the main factor affecting runtime. Table S1 shows the runtime over other three numerical terrains. The results similar to Table 4 prove that the efficiency of any flow distance algorithm is steady and its runtime is mainly determined by the size of the DEM.

The computation efficiency of MEB is also acceptable with less than

Table 3
Relative errors between the exact length and the predicted river lengths using the equidistant belt areas by different flow distance algorithms and width function extraction modes.

	D8-CL-SEB	D8-CL-MEB	D8-CT-SEB	D8-CT-MEB	iFAD8-CT-SEB	iFAD8-CT-MEB	iFAD8-M-SEB	iFAD8-M-MEB	D∞-TLI-SEB	D∞-TLI-MEB
Width at distance of 1.5 m (m)	17683	15425	18514	15779	18343	16129	17633	15682	18222	15471
Width at distance of 2.5 m (m)	19731	16118	18123	16546	17994	16745	21262	18300	17528	16277
Width at distance of 3.5 m (m)	17288	16091	16198	16521	16792	16770	17065	16840	17538	16278
Predicted river length (m)	8842	7712	9257	7889	9172	8064	8817	7841	9111	7736
Relative error (%)	17.17	2.21	22.67	4.55	21.54	6.87	16.84	3.91	20.74	2.51

Table 4
The runtimes of different flow distance algorithms to process the plane with different resolutions.

Resolution (m)	Cell Numbers	D8-CL	D8-DT	D8-CT	iFAD8-CL	iFAD8-DT	iFAD8-CT	iFAD8-M	QMFD-CL	D∞-CL	D∞-TLI
30	1.02×10^4	0.003	0.004	0.007	0.012	0.015	0.017	0.01	0.005	0.008	0.007
20	2.28×10^4	0.01	0.008	0.017	0.027	0.025	0.037	0.02	0.009	0.022	0.018
10	9.06×10^4	0.043	0.042	0.064	0.113	0.108	0.137	0.093	0.046	0.068	0.067
5	3.61×10^5	0.198	0.227	0.324	0.513	0.551	0.632	0.459	0.269	0.33	0.318
2	2.25×10^6	2.097	2.155	2.877	4.379	4.583	5.181	5.554	2.378	2.775	2.895
1	9.01×10^6	11.562	10.306	16.936	21.513	24.577	27.01	59.79	13.971	17.333	18.401

Note. The unit is in second. The evaluation is performed on a computer with an Intel Xeon E5-1620 v4 CPU and 80 GB of memory.

14.0 s required to process any 1 m-resolution DEM using any given flow distance distribution. This runtime is much longer than the SEB algorithm (0.08 s), but is shorter than the runtimes of most flow distance algorithms.

4.4. Suggestions for further studies

The algorithms are assessed over four numerical terrains and two real-world terrains. The selected numerical terrains are typical and can represent most of the basic topographic units (Li et al., 2021). So the conclusions based on experiments over numerical terrains are convictive, and the results over real-world terrains support these conclusions. However, although the difference between the accuracies of different flow distance algorithms or width function algorithms is obvious, it is still not clear whether this difference can greatly affect the result of hydrological modeling. Therefore, it is necessary to apply these algorithms to runoff simulation for a better comparison.

In addition, the new algorithm selects TLI because its straight isolines are suitable for the calculation of equidistant belt area. This is a compromise to practicability, but some other interpolation methods may have better precision (Xu et al., 2018). Hence, in order to improve the precision, more attempts can be taken to couple other interpolation methods into the structure of D∞-TLI.

The execution time of the new algorithms seems to be acceptable, but there is potential for further improvement. For example, D∞-TLI uses a similar structure to priority-flood, and Barnes et al. (2014) have proposed a method to decrease the time complexity of priority-flood to $O(n)$. For MEB, while each cell is processed independently, parallel computing is a strategy that can be adopted to improve computing efficiency. If these strategies can be included, the practicability of the new algorithm will undoubtedly be further improved.

5. Concluding remarks

Flow distance and width function are important hydrological parameters. However, it is difficult for existing algorithms to extract them accurately, and a major problem limiting the accuracy of the algorithm is the lack of flow distance values in a cell region except for the cell center. Hence, this study adopts TLI to provide the missing values and proposes a new method. This method can be divided into two parts. One part can estimate the flow distance distribution based on grid DEMs, and the other part can calculate the width function using the estimated flow distance distribution. The new flow distance algorithm (D∞-TLI) adopts a two-segmented-distance strategy that divides the flow distance into two segments whose sub-distances are approximated using D∞ and TLI, respectively. Then, the continuous flow distance field is approximated over each cell region, so this cell area can be divided into multiple equidistant belts (MEB) for the width function.

Four numerical terrains and two real-world terrains with multiple resolutions are adopted for assessments. The results indicate that D∞-TLI generally outperforms nine existing flow distance algorithms and causes low average *MARE*s of 2.31 % and 4.01% for the estimated flow distance distribution over the numerical and the real-world terrains, respectively. Compared with the traditional method, MEB can effectively decrease the artificial fluctuations in the estimated width function. The combination of D∞-TLI with MEB (i.e., D∞-TLI-MEB) outperforms other combinations by providing estimated width functions with an average *MARE* of 2.97 % for the numerical terrains, while it also works well over real-world terrains. Except for the average *MARE*, the high match between the estimated values and the exact values shows that D∞-TLI and D∞-TLI-MEB has better accuracy for most positions in any terrain (Figs. 5 and 9). Due to the structure similar to priority-flood, the complex D∞-TLI method maintains high operating efficiency. Moreover, MEB has a simpler structure and higher efficiency. Therefore, D∞-TLI and MEB can provide hydro-geomorphological attributes with higher accuracy and efficiency. So these methods have great potential in hydrological and geomorphological applications.

Software and data availability

The codes of the proposed algorithms to extract the flow distance and the width function are programmed by java language (version 1.8). These codes are available at a Figshare repository created by Pengfei Wu (E-mail: wpf@hhu.edu.cn) in 2023: https://doi.org/10.6084/m9.figshare.22010132.v2. This repository contains the program codes (40 KB) and the sample data (443 KB). Author's experimental environment was as follows:

- OS: Windows 10 Pro.
- CPU: Intel(R) Xeon(R) E5-1620 v4 3.50 GHz.
- RAM: 80.00 GB.
- GPU: NVIDIA Quadro M2000

The DEMs of the numerical terrains are provided by Wu et al. (2022). The LiDAR DEM containing the SCT Basin is distributed by OpenTopography at: https://doi.org/10.5069/G9RB72JV. The ALOS DEM containing the DDG Basin is available at: https://search.earthdata.nasa.gov. The real-world basin domains as well as the mapped channels are available at a Figshare repository created by Pengfei Wu in 2023: https://doi.org/10.6084/m9.figshare.22004444.v1.

CRediT authorship contribution statement

Pengfei Wu: Conceptualization, Methodology, Software, Validation, Writing - original draft, Visualization, Data curation, Funding acquisition. Jintao Liu: Methodology, Supervision, Funding acquisition, Writing - review & editing, Project administration. Meiyan Feng: Validation, Writing - review & editing. Hu Liu: Validation, Writing - review & editing.

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.

Data availability

A detailed software and data availability section is provided. In this section, the links of all the codes and data are listed.

Acknowledgements

This work was supported by the Second Tibetan Plateau Scientific Expedition and Research Program (STEP; grant no. 2019QZKK0207-02), the National Natural Science Foundation of China (NSFC; grant no. 92047301), the Postgraduate Research & Practice Innovation Prograikm of Jiangsu Province (grant no. KYCX22_0636), and the Fundamental Research Funds for the Central Universities (grant no. B230205038). The authors are grateful to Dan Ames and two anonymous reviewers for valuable suggestions.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.envsoft.2023.105865.

Algorithm 1. The proposed width function algorithm. The array WF is the estimated width function. If only the flow distance distribution (i.e., the matrix FD) is required, the algorithm can be stopped at the end of Line 21.

Require non-depression *DEM*, equidistant belt interval (*EBI*), rasterized *Channel* layer and *Hillslope* layer

1: Let *PQ* be a min-first priority queue
2: Let *FD* have the same dimensions as *DEM*
3: Let *IN* have the same dimensions as *DEM*
4: Let *IN* be initialized to FALSE
5: **for all** *c* of *DEM* **do**
6: **if** *c* belongs to *Channel* or the edge of *DEM* **then**
7: *FD*(*c*)←0
8: *IN*(*c*) ←TRUE
9: **for all** neighbors *n* of *c* **do**
10: **if** *IN*(*n*) **then repeat loop**
11: Push *n* onto *PQ* with priority *DEM*(*n*)
12: *IN*(*n*)←TRUE
13: **while** *PQ* is not empty **do**
14: *c*←POP(*PQ*)
15: Let n_c be the closest cardinal neighbor to the D∞ direction of *c*
16: Let n_d be the closest diagonal neighbor to the D∞ direction of *c*
17: *FD*(*c*) ←*Equation 5* (*FD*(n_c), *FD*(n_d))
18: **for all** neighbors *n* of *c* **do**
19: **if** IN(n) **then repeat loop**
20: Push *n* onto *PQ* with priority *DEM*(*n*)
21: *IN*(*n*)←TRUE
22: Let *EB* be a one-dimensional array of the length equal to *ceiling*(*max*(*FD*)/*EBI*)
23: Let *WF* be an array of the same size as *EB*
24: **for all** *c* of *Hillslope* **do**
25: **for all** facets *f* of *c* **do**
26: **for all** equidistant belt *e* of *f* **do**
27: *EB*(*ceiling*(*e.elev*/*EBI*))= *EB*(*ceiling*(*e.elev*/*EBI*))+*e.area*
28: **for all** elements *m* of *EB* **do**
29: *WF*(*m*)=*EB*(*m*)/*EBI*

References

Barnes, R., Lehman, c., Mulla, D., 2014. Priority-flood: an optimal depression-filling and watershed-labeling algorithm for digital elevation models. Comput. Geosci. 62, 117–127. https://doi.org/10.1016/j.cageo.2013.04.024.

Blodgett, D., Johnson, J.M., Sondheim, M., Wieczorek, M., Frazier, N., 2021. Mainstems: a logical data model implementing mainstem and drainage basin feature types based on WaterML2 Part 3: HY Features concepts. Environ. Model. Software 135, 104927. https://doi.org/10.1016/j.envsoft.2020.104927.

Bogaart, B.W., Troch, P.A., 2006. Curvature distribution within hillslopes and catchments and its effect on the hydrological response. Hydrol. Earth Syst. Sci. 10, 925–936. https://doi.org/10.5194/hess-10-925-2006.

Butt, M.A., Maragos, P., 1998. Optimum design of chamfer distance transforms. IEEE Trans. Image Process. 7 (10), 1477–1484. https://doi.org/10.1109/83.718487.

Chantaveeroda, A., Woradit, K., Seagar, A., Limpiti, T., 2023. Novel catchment estimation for super-resolution DEM with physically based algorithms: surface water path delineation and specific catchment area calculation. IEEE Access 11, 3293399. https://doi.org/10.1109/ACCESS.2023.3293399.

De Smith, M., 2004. Distance transforms as a new tool in spatial analysis, urban planning, and GIS. Environ. Plann. Plann. Des. 31 (1), 85–104. https://doi.org/10.1068/b29123.

Di Lazzaro, M., Zarlenga, A., Volpi, E., 2016. Understanding the relative role of dispersion mechanisms across basin scales. Adv. Water Resour. 91, 23–36. https://doi.org/10.1016/j.advwatres.2016.03.003.

Dong, L., Ge, C., Zhang, H., Liu, Z., Yang, Q., Jin, B., Ritsema, C.J., Geissen, V., 2022. An optimized method for extracting slope length in RUSLE from raster digital elevation. Catena 209, 105818. https://doi.org/10.1016/j.catena.2021.105818.

Fan, Y., Bras, R.L., 1998. Analytical solutions to hillslope subsurface storm flow and saturation overland flow. Water Resour. Res. 34, 921–927. https://doi.org/10.1029/97WR03516.

Fan, F.M., Fleischmann, A.S., Collischonn, W., Ames, D.P., Rigo, D., 2015. Large-scale analytical water quality model coupled with GIS for simulation of point sourced pollutant discharges. Environ. Model. Software 64, 58–71. https://doi.org/10.1016/j.envsoft.2014.11.012.

Farahbakhsh, F., Shahidinejad, A., Ghobaei-Arani, M., 2020. Multiuser context-aware computation offloading in mobile edge computing based on Bayesian learning automata. Trans. Emerging Tel. Tech. 32, e4127. https://doi.org/10.1002/ett.4127.

Fei, J., Liu, J., Ke, L., Wang, W., Wu, P., Zhou, Y., 2022. A deep learning-based method for mapping alpine intermittent rivers and ephemeral streams of the Tibetan Plateau from Sentinel-1 time series and DEMs. Remote Sens. Environ. 282, 113271 https://doi.org/10.1016/j.rse.2022.113271.

Gupta, V., Waymire, E., Rodíguez-Iturbe, I., 1986. On scales, gravity and network structure in basin runoff. In: Gupta, V., Waymire, E., Rodíguez-Iturbe, I. (Eds.), Scale Problems in Hydrology. D. Reidel Publishing Co., pp. 159–184. https://doi.org/10.1007/978-94-009-4678-1_8

Hazenberg, P., Fang, Y., Broxton, P., Gochis, D., Niu, G.Y., Pelletier, J.D., Troch, P.A., Zeng, X., 2015. A hybrid-3D hillslope hydrological model for use in Earth system models. Water Resour. Res. 51, 8218–8239. https://doi.org/10.1002/2014WR016842.

Hickey, R., 2000. Slope angle and slope length solutions for GIS. Cartography 29 (1), 1–8. https://doi.org/10.1080/00690805.2000.9714334.

Hooshyar, M., Wang, D., Kim, S., Medeiroset, S.C., Hagen, S.C., 2016. Valley and channel networks extraction based on local topographic curvature and k-means clustering of contours. Water Resour. Res. 52, 8081–8102. https://doi.org/10.1002/2015WR018479.

Kotyra, B., Chabudziński, Ł., 2023. Fast parallel algorithms for finding the longest flow paths in flow direction grids. Environ. Model. Software 167, 105728. https://doi.org/10.1016/j.envsoft.2023.105728.

Lapides, D., Sytsma, A., O'Neil, G., Djokic, D., Nichols, M., Thompson, S., 2022. Arc hydro hillslope and critical duration: new tools for hillslope-scale runoff analysis. Environ. Model. Software 153, 105408. https://doi.org/10.1016/j.envsoft.2022.105408.

Li, T., Wang, G., Chen, J., Wang, H., 2011. Dynamic parallelization of hydrological model simulations. Environ. Model. Software 26 (12), 1736–1746. https://doi.org/10.1016/j.envsoft.2011.07.015.

Li, J., Li, T., Zhang, L., Sivakumar, B., Fu, X., Huang, Y., Bai, R., 2020. A D8-compatible high-efficient channel head recognition method. Environ. Model. Software 125, 104624. https://doi.org/10.1016/j.envsoft.2020.104624.

Li, Z., Yang, T., Wang, C., Shi, P., Yong, B., Song, Y., 2021. Assessing the precision of total contributing area (TCA) estimated by flow direction algorithms based on the analytical solution of theoretical TCA on synthetic surfaces. Water Resour. Res. 57, e2020WR028546 https://doi.org/10.1029/2020WR028546.

Liu, J., Chen, X., Zhang, X., Hoagland, K.D., 2012. Grid digital elevation model based algorithms for determination of hillslope width functions through flow distance transforms. Water Resour. Res. 48, W04532 https://doi.org/10.1029/2011WR011395.

Liu, J., Han, X., Chen, X., Lin, H., Wang, A., 2016. How well can the subsurface storage-discharge relation be interpreted and predicted using the geometric factors in headwater areas? Hydrol. Process. 30 (25), 4826–4840. https://doi.org/10.1002/hyp.10958.

Masdari, M., Gharehpasha, S., Ghobaei-Arani, M., Ghasemi, V., 2020. Bio-inspired virtual machine placement schemes in cloud computing environment: taxonomy, review, and future research directions. Cluster Comput. 23, 2533–2563. https://doi.org/10.1007/s10586-019-03026-9.

Maxwell, J.C., 1870. On hills and dales. Philos. Mag. J. Sci. 40 (269), 421–427.

Mayorga, E., Logsdon, M.G., Ballester, M., Richey, J.E., 2005. Estimating cell-to-cell land surface drainage paths from digital channel networks, with an application to the amazon basin. J. Hydrol. 315, 167–182. https://doi.org/10.1016/j.jhydrol.2005.03.023.

McGuire, K.J., McDonnell, J.J., Weiler, M., Kendall, C., McGlynn, B.L., Welker, J.M., Seibert, J., 2005. The role of topography on catchment-scale water residence time. Water Resour. Res. 41, W05002 https://doi.org/10.1029/2004WR003657.

Moussa, R., 2008. What controls the width function shape, and can it be used for channel network comparison and regionalization? Water Resour. Res. 44, W08456 https://doi.org/10.1029/2007WR006118.

Mutzner, R., Tarolli, P., Sofia, G., Parlange, M.B., Rinaldo, A., 2016. Field study on drainage densities and rescaled width functions in a high-altitude alpine catchment. Hydrol. Process. 30, 2138–2152. https://doi.org/10.1002/hyp.10783.

Noël, P., Rousseau, A.N., Paniconi, C., Nadeau, D.F., 2014. Algorithm for delineating and extracting hillslopes and hillslope width functions from gridded elevation data. J. Hydrol. Eng. 19 (2), 366–374. https://doi.org/10.1061/(ASCE)HE.1943-5584.0000783.

O'Callaghan, J.F., Mark, D.M., 1984. The extraction of drainage networks from digital elevation data. Comput. Vis. Graph Image Process 28 (3), 323–344. https://doi.org/10.1016/S0734-189X(84)80011-0.

Orlandini, S., Moretti, G., Franchini, M., Aldighieri, B., Testa, B., 2003. Path-based methods for the determination of nondispersive drainage directions in grid-based digital elevation models. Water Resour. Res. 39 (6), 1144. https://doi.org/10.1029/2002WR001639.

Orlandini, S., Moretti, G., Corticelli, M.A., Santangelo, P.E., Capra, A., Rivola, R., Albertson, J.D., 2012. Evaluation of flow direction methods against field observations of overland flow dispersion. Water Resour. Res. 48, W10523 https://doi.org/10.1029/2012WR012067.

Orlandini, S., Moretti, G., Gavioli, A., 2014. Analytical basis for determining slope lines in grid digital elevation models. Water Resour. Res. 50, 526–539. https://doi.org/10.1002/2013WR014606.

Paik, K., 2008. Global search algorithm for nondispersive flow path extraction. J. Geophys. Res. 113, F04001 https://doi.org/10.1029/2007JF000964.

Paz, A.R., Collischonn, W., Risso, A., Mendes, C.A.B., 2008. Errors in river lengths derived from raster digital elevation models. Comput. Geosci. 34, 1584–1596. https://doi.org/10.1016/j.cageo.2007.10.009.

Pilesjö, P., Hasan, A., 2014. A triangular form-based multiple flow algorithm to estimate overland flow distribution and accumulation on a digital elevation model. T. GIS 18 (1), 108–124. https://doi.org/10.1111/tgis.12015.

Quinn, P., Beven, K., Chevallier, P., Planchon, O., 1991. The prediction of hillslope flow paths for distributed hydrological modelling using digital terrain models. Hydrol. Process. 5, 59–79. https://doi.org/10.1002/hyp.3360050106.

Ranjram, M., Craig, J.R., 2021. Use of an efficient proxy solution for the hillslope-storage Boussinesq problem in upscaling of subsurface stormflow. Water Resour. Res. 57, e2020WR029105 https://doi.org/10.1029/2020WR029105.

Rigon, R., Bancheri, M., Formetta, G., de Lavenne, A., 2016. The geomorphological unit hydrograph from a historical-critical perspective. Earth Surf. Process. Landforms 41, 27–37. https://doi.org/10.1002/esp.3855.

Sahoo, S., Sahoo, B., 2019a. A geomorphology-based integrated stream-aquifer interaction model for semi-gauged catchments. Hydrol. Process. 33, 1362–1377. https://doi.org/10.1002/hyp.13406.

Sahoo, S., Sahoo, B., 2019b. Modelling the variability of hillslope drainage using grid-based hillslope width function estimation algorithm. ISH J. Hydraul. Eng. 25 (1), 71–78. https://doi.org/10.1080/09715010.2018.1441750.

Schwendel, A.C., Fuller, I.C., Death, R.G., 2012. Assessing DEM interpolation methods for effective representation of upland stream morphology for rapid appraisal of bed stability. River Res. Appl. 28 (5), 567–584. https://doi.org/10.1002/rra.1475.

Shin, S., Paik, K., 2017. An improved method for single flow direction calculation in grid digital elevation models. Hydrol. Process. 31 (8), 1650–1661. https://doi.org/10.1002/hyp.11135.

Sloan, S.W., 1987. A fast algorithm for constructing Delaunay triangulations in the plane. Adv. Eng. Software 9 (1), 34–55.

Tarboton, D.G., 1997. A new method for the determination of flow directions and upslope areas in grid digital elevation models. Water Resour. Res. 33 (2) https://doi.org/10.1029/96WR03137, 662-319.

Tesfa, T.K., Tarboton, D.G., Chandler, D.G., McNamara, J.P., 2009. Modeling soil depth from topographic and land cover attributes. Water Resour. Res. 45, W10438 https://doi.org/10.1029/2008WR007474.

Troch, P.A., Paniconi, C., van Loon, E.E., 2003. Hillslope-storage Boussinesq model for subsurface flow and variable source areas along complex hillslopes: 1. formulation and characteristic response. Water Resour. Res. 39 (11), 1316. https://doi.org/10.1029/2002WR001728.

Van Nieuwenhuizen, N., Lindsay, J.B., DeVries, B., 2021. Smoothing of digital elevation models and the alteration of overland flow path length distributions. Hydrol. Process. 35 (7), e14271 https://doi.org/10.1002/hyp.14271.

Veneziano, D., Moglen, G.E., Furcolo, P., Iacobelli, V., 2000. Stochastic model of the width function. Water Resour. Res. 36 (7), 1143–1157. https://doi.org/10.1029/2000WR900002.

Wilson, J.P., Lam, C.S., Deng, Y., 2007. Comparison of the performance of flow-routing algorithms used in GIS-based hydrologic analysis. Hydrol. Process. 21, 1026–1044. https://doi.org/10.1002/hyp.6277.

Wu, P., Liu, J., Han, X., Liang, Z., Liu, Y., Fei, J., 2020. Nondispersive drainage direction simulation based on flexible triangular facets. Water Resour. Res. 56, e2019WR026507 https://doi.org/10.1029/2019WR026507.

Wu, P., Liu, J., Han, X., Feng, M., Fei, J., Shen, X., 2022. An improved triangular form-based multiple flow direction algorithm for determining the nonuniform flow domain over grid networks. Water Resour. Res. 58, e2021WR031706 https://doi.org/10.1029/2021WR031706.

Xu, H., Ma, C., Lian, J., Xu, K., Evance, C., 2018. Urban flooding risk assessment based on an integrated k-means cluster algorithm and improved entropy weight method in the region of Haikou, China. J. Hydrol. 563, 975–986. https://doi.org/10.1016/j.jhydrol.2018.06.060.

Yilmaz, H.M., 2007. The effect of interpolation methods in surface definition: an experimental study. Earth Surf. Process. Landforms 32, 1346–1361. https://doi.org/10.1002/esp.1473.

Zanella-Béguelin, S., Wutschitz, L., Tople, S., Salem, A., Rühle, V., Paverd, A., Naseri, M., Köpf, B., Jones, D., 2023. Bayesian estimation of differential privacy. In: Proceedings of the 40th International Conference on Machine Learning. https://doi.org/10.48550/arXiv.2206.05199. PMLR 211.

Zhang, Y., Zhang, C., Chen, H., Grieneisen, M.L., Liu, J., Zhang, M., 2016. Spatial interpolation of river channel topography using the shortest temporal distance. J. Hydrol. 542, 450–462. https://doi.org/10.1016/j.jhydrol.2016.09.022.

Zhou, Q., Pilesjö, P., Chen, Y., 2011. Estimating surface flow paths on a digital elevation model using a triangular facet network. Water Resour. Res. 47, W07522 https://doi.org/10.1029/2010WR009961.