Journal of Mathematical Imaging and Vision (2018) 60:1025–1036
https://doi.org/10.1007/s10851-018-0793-1

Path-Value Functions for Which Dijkstra's Algorithm Returns Optimal Mapping

Krzysztof Chris Ciesielski[1,2] · Alexandre Xavier Falcão[3] · Paulo A. V. Miranda[4]

Received: 12 September 2017 / Accepted: 17 January 2018 / Published online: 3 February 2018

Abstract

Dijkstra's algorithm (DA) is one of the most useful and efficient graph-search algorithms, which can be modified to solve many different problems. It is usually presented as a tool for finding a mapping which, for every vertex v, returns a shortest-length path to v from a fixed single source vertex. However, it is well known that DA returns also a correct optimal mapping when multiple sources are considered and for path-value functions more general than the standard path-length. The use of DA in such general setting can reduce many image processing operations to the computation of an optimum-path forest with path-cost function defined in terms of local image attributes. In this paper, we describe the general properties of a path-value function defined on an arbitrary finite graph which, provably, ensure that Dijkstra's algorithm indeed returns an optimal mapping. We also provide the examples showing that the properties presented in a 2004 TPAMI paper on the image foresting transform, which were supposed to imply proper behavior of DA, are actually insufficient. Finally, we describe the properties of the path-value function of a graph that are provably necessary for the algorithm to return an optimal mapping.

Keywords Dijkstra's algorithm · Graph-search algorithms · Image foresting transform · Connectivity functions

1 Introduction

In 1959, Edsger Wybe Dijkstra presented a note [18] describing the solutions of two graph-search problems for connected edge-weighted graphs. The solution of the second problem, on finding the shortest-length path from a single source vertex s to another vertex v, can be trivially extended to multiple sources, that is, given a non-empty subset S of vertices, to finding a path from a $u \in S$ to v whose length does not exceed the length of any other path from a $w \in S$ to v. This extension can be done by adding to the original graph the dummy vertex s, connecting it to each vertex in S by an edge of weight zero, and finding for this extended graph the shortest-length path from s to v. Ever since, the solution of the shortest path problem from [18] is known as *Dijkstra's algorithm*. It has been applied, in the original or a modified form, in the multitude of different practical tasks, like routing phone calls in telephone networks, finding the best flights between airports for a given departure time, and designating file servers in the local computer networks.

The modified versions of Dijkstra's algorithm usually rely on some monotone path-value function [24], and they can either minimize or maximize an optimum path-value map. In [21] the authors proposed the *image foresting transform, IFT*, a methodology to design image processing operators based on the modifications of Dijkstra's algorithm to multiple sources and more general path-value functions. The IFT essentially reduces image processing operators to the computation of an optimum-path forest in a graph derived from the image, followed by a local processing of its attributes. Its applications include boundary-based [22,23,29], region-based [1,4,5,15,16,25,27,28,41,42], and hybrid [10,37] image segmentation, morphological reconstructions [20], simultaneous connected filtering and watershed transforms [26], fast binary morphology [33], linear time exact Euclidean distance transform and one-pixel-wide connected multiscale skeletonization [19],

✉ Krzysztof Chris Ciesielski
KCies@math.wvu.edu

[1] Department of Mathematics, West Virginia University, Morgantown, WV 26506-6310, USA

[2] MIPG, Department of Radiology, University of Pennsylvania, Philadelphia, PA 19104-6021, USA

[3] Institute of Computing, University of Campinas, Campinas, SP 13083-852, Brazil

[4] Institute of Mathematics and Statistics, University of São Paulo, São Paulo, SP 05508-090, Brazil

shape description [3,39,40], clustering [6,35], and classification [2,31,32,36].

In this paper, we describe the properties that a path-value function of a graph must satisfy in order to ensure that Dijkstra's algorithm returns an optimum path-value map, provide examples to show that the path-value function properties presented in [21] are insufficient to ensure the "proper" behavior of the algorithm, and present a simple variant of Dijkstra's algorithm that guarantees its output to be a spanning forest. Note that the published IFT-based image operators either satisfy the sufficient condition [5,6,15,19,20,23,26,27,31,41] or have been proposed by using the aforementioned variant that guarantees a spanning forest [1,11,25,27,29,38]. Therefore, the main contribution of this work is the formulation of the general conditions of the path-value functions that provably ensure that the algorithm returns an optimum path-value map.

This paper is organized as follows: Sect. 2 presents the basic definitions and notation, with examples of the most commonly used path-value functions, especially in image processing. The characterization theorem for Dijkstra's algorithm and its aforementioned variant are presented in Sects. 3 and 4. Section 5 shows that the properties given in [21] as sufficient for ensuring that the algorithm works correctly are actually insufficient. Comments on optimization and the proofs are presented in Sects. 6 and 7, respectively. Conclusions are stated in Sect. 8.

2 Basic Definitions and Examples Pertinent to the Algorithm

Let $G = \langle V, E\rangle$ be a directed graph, where V is a non-empty finite set of its vertices and $E \subset V \times V$ is the set of its edges. We assume also that G has no loops, that is, $\langle v, v\rangle \notin E$ for every $v \in V$. A *path* (*in* G), with *terminus* $v = v_\ell$ and of length $\ell \geq 0$, is any sequence $p_v = \langle v_0, \dots, v_\ell\rangle$ of vertices such that $\langle v_j, v_{j+1}\rangle \in E$ for any $j < \ell$; it is *from* $S \subset V$ *to* $v \in V$ when $v_0 \in S$ and $v_\ell = v$; and if $\langle v, w\rangle \in E$, then $p_v\hat{}w$ denotes the path $\langle v_0, \dots, v_\ell, w\rangle$. Let Π_G be the family of all paths in G and consider a path-value function $\psi \colon \Pi_G \to [-\infty, \infty]$, where $[-\infty, \infty]$—the extended real line—is considered with the curly order relation $\psi(p_v) \preceq \psi(q_v)$, being either $\psi(p_v) \leq \psi(q_v)$ or $\psi(p_v) \geq \psi(q_v)$. The choice of $\preceq$ as either $\leq$ or $\geq$ depends on the application, and it is always clear from the context.

Commonly, the path-value function ψ is defined from an edge-weight map $\omega_E \colon E \to \mathbb{R}$ (i.e., G is an edge-weighted graph $G = \langle V, E, \omega_E\rangle$) which, in different applications, is referred to as the *local distance*, *cost*, or *affinity* function, see Examples 1 and 2. Also, in some cases, ψ is defined from a vertex-weight map $\omega_V \colon V \to \mathbb{R}$, see Examples 3 and 4. However, we assume here only that the function ψ is computable by a readily available algorithm. In particular, the definition of ψ need not depend on either edge- or vertex-weight map, see Examples 6 and 10. Also, in general, a set $S \subset V$ of seeds (i.e., of vertices where all cost-effective paths must start) need not be specified, see Examples 3, 6, and 7.

A Dijkstra-type algorithm associated with ψ is concerned with finding, for every $v \in V$, the cost/strength $\psi(p_v)$ of a *ψ-optimal path* in Π_G to v. We say that a map $\sigma \colon V \to [-\infty, \infty]$ is a *ψ-optimal map* provided, for every $v \in V$, $\sigma[v] = \psi(p_v)$ for some ψ-optimal path p_v to v. Since "optimal" may mean either standard-order minimal (e.g., as in Example 1) or standard-order maximal (e.g., as in Example 2), we will define $\preceq$ as $\leq$ in the former case and as $\geq$ in the latter case. This will allow us to talk uniformly on the $\preceq$-minimization task, independently on which of the two situations we consider.

Typically, a Dijkstra-type algorithm actually finds a map $\pi \colon V \to \Pi_G$ such that, for every $v \in V$, $\pi[v]$ is a path to v. This map induces the map $\sigma \colon V \to [-\infty, \infty]$ as a composition $\sigma = \psi \circ \pi$, that is, σ is given via formula $\sigma[v] = \psi(\pi[v])$. The family $P = \{\pi[v] \colon v \in V\}$ usually forms a *forest* in the graph, that is, it has the properties: (i) For every $v \in V$ there exists a unique path $p_v \in P$ to v; (ii) every initial segment of a $p = \langle v_0, \dots, v_\ell\rangle \in P$ (i.e., $\langle v_0, \dots, v_k\rangle$ for $k \leq \ell$) also belongs to P.

Most applications define the cost map ψ so that all optimal paths must start from an explicitly given non-empty set $S \subset V$ of *seeds*. In such cases, the fact that all ψ-optimal paths indeed start at S is ensured by requiring that $\psi(p_v) \prec \psi(q_v)$ whenever $p_v \in \Pi_G$ is from S and $q_v \in \Pi_G$ is from $V \backslash S$. See examples of commonly used path-value functions below. However, in what follows, we do not require that the cost functions are defined with an explicitly specified seed set. (However, one can always consider the entire set V as being a set of seeds.) In image processing, $G = \langle V, E\rangle$ is commonly a grid graph, with V being the image domain and the set E of edges the connectors of adjacent pixels.

Example 1 The classic Dijkstra's shortest path algorithm [18] searches, for every $v \in V$, for a path from a fixed non-empty set $S \subset V$ of seeds to v of *minimal* weighted length. So, $\preceq$ is interpreted as $\leq$. The algorithm uses the path-value function—the length—defined from the local distance $\omega_E \colon E \to [0, \infty)$ as follows. Whenever $v_0 \in S$ we put $\psi_{\text{sum}}(\langle v_0, \dots, v_\ell\rangle) = \sum_{1 \leq j \leq \ell} \omega_E(v_{j-1}, v_j)$ for $\ell > 0$ and $\psi_{\text{sum}}(\langle v_0\rangle) = 0$; otherwise, $\psi_{\text{sum}}(\langle v_0, \dots, v_\ell\rangle) = \infty$.

In image processing, an optimum contour tracking operation constrained to a set of strokes across the object's boundary, geodesic dilations of a binary set, and some approaches for region-based image segmentation [1,4,21–23] rely on the modifications of ω_E in ψ_{sum} for one or multiple sources.

Example 2 Another example is the path-value function $\psi_{\min}$. In the *fuzzy connectedness* applications [7–9,12–15,25,41,42], this function $\psi_{\min}$ is used to measure the "strength of connectivity" between vertices, as a function of a local connectivity (i.e., *affinity*) map $\omega_E\colon E \to [0,1]$ defined as follows. If $v_0 \in S$, then

$$\psi_{\min}(\langle v_0,\dots,v_\ell\rangle) = \min_{1\le j\le\ell} \omega_E(v_{j-1},v_j) \in [0,1]$$

for $\ell > 0$ and $\psi_{\min}(\langle v_0\rangle) = 1$. Otherwise, for $v_0 \notin S$, we put $\psi_{\min}(\langle v_0,\dots,v_\ell\rangle) = -\infty$. Its applications are concerned with the paths of *maximal* strength of connectedness. So, for $\psi_{\min}$, we will interpret $\preceq$ as $\geq$.

Example 3 A path-value function can also be defined from a vertex *altitude* map $\omega_V\colon V \to [-\infty,\infty)$ via formula $\psi_{\text{peak}}(\langle v_0,\dots,v_\ell\rangle) = \max_{1\le j\le\ell}\{h(v_0),\omega_V(v_j)\}$ for $\ell > 0$, and $\psi_{\text{peak}}(\langle v_0\rangle) = h(v_0)$ for some handicap value $h(v_0) \geq \omega_V(v_0)$ for all $v_0 \in V$. In image processing, its applications involve superior morphological reconstructions and watershed transforms [20,21,26], which are concerned with the paths of *minimal* peak. So, for ψ_{peak}, we will interpret $\preceq$ as $\leq$.

The handicap values may be defined as $h(v_0) = \omega_V(v_0)$ for $v_0 \in S$, and $h(v_0) = \infty$ otherwise, as is the case in the watershed transform from a set S of labeled markers [21]. For superior reconstruction [20], we may define $h \geq \omega_V$ and, for watershed transforms from a grayscale marker, $h > \omega_V$ as discussed in [26]. In both cases, the set S can only be discovered on the fly, as derived from the minima of the resulting ψ_{peak}-*minimal* map $\sigma[\,]$.

Example 4 Another example of a path-value function based on a vertex *altitude* map $\omega_V\colon V \to [0,\infty)$ is defined as follows. If $v_0 \in S$, then

$$\psi_{\text{dif}}(\langle v_0,\dots,v_\ell\rangle) = \max_{0\le j\le\ell} \omega_V(v_j) - \min_{0\le j\le\ell} \omega_V(v_j)$$

for $\ell > 0$ and $\psi_{\text{dif}}(\langle v_0\rangle) = 0$. Otherwise, for $v_0 \notin S$, $\psi_{\text{dif}}(\langle v_0,\dots,v_\ell\rangle) = \infty$. Its applications are concerned with the paths of *minimal* height (difference between maximum and minimum altitudes). So, for ψ_{dif}, we will interpret $\preceq$ as $\leq$.

In image processing, ψ_{dif} defines a *minimal barrier distance* between vertices, which is useful in some image segmentation applications [11,38].

Example 5 Yet another example of a path-value function based on a vertex *altitude* map $\omega_V\colon V \to [0,\infty)$ is the map $\psi_{\text{last}}\colon V \to [0,\infty)$, which is defined as $\psi_{\text{last}}(\langle v_0,\dots,v_\ell\rangle) = \omega_V(v_\ell)$ when $v_0 \in S$ and as $\psi_{\text{last}}(\langle v_0,\dots,v_\ell\rangle) = \infty$ otherwise. Its applications seek the paths of *minimal* strength. Examples are a particular case of the riverbed boundary tracking [29] and the imposition of connectivity constraints in region-based image segmentation [28]. Thus, for ψ_{last}, we will interpret $\preceq$ as $\leq$.

3 Dijkstra's Algorithm (DA) and the Correctness Theorem

In the following algorithm, any time during its execution and for any $v \in V$, $\pi[v]$ is a path to v with $\sigma[v] = \psi(\pi[v])$. The algorithm, putting aside notational differences, is identical to the one studied in [21] with a minor exception—the paths that we store in the array $\pi[\,]$ were indicated in [21] via predecessor map $P[\,]$: A path $\pi[v_\ell] = \langle v_0 \dots, v_\ell\rangle$ was indicated through the assignments $P[v_0] = nil$ and $P[v_i] = v_{i-1}$ for any $i \in \{1,\dots,\ell\}$. In the algorithm, we use the operation arg $\preceq$-opt that finds a vertex v in H for which the value of $\psi(\pi[v])$ has a $\preceq$-minimal value, that is, consists of the (standard) minimum value in case of standard-order minimization and the (standard) maximum value in case of standard-order maximization.

Algorithm 1: Dijkstra's algorithm **DA**, aiming to find the ψ-optimal map

```
Data: A finite graph $G = \langle V, E\rangle$ and a path-value function $\psi$
      from $\Pi_G$ to $\langle[-\infty,\infty], \preceq\rangle$
Result: An array $\sigma[\,]$ of numbers, aiming for being $\psi$-optimal
        map
Additional Structure: A variable $\sigma'$, a set H, and an
array $\pi[\,]$ of paths, such that, at any time and for any $v \in V$,
$\pi[v]$ is a path to $v$ with $\sigma[v] = \psi(\pi[v])$
1 foreach v $\in V$ do $\pi$[v] $\leftarrow$ ⟨v⟩; $\sigma$[v] $\leftarrow \psi(\pi$[v]$)$
                                        /* initialization loop */
2 H $\leftarrow V$
3 while H $\neq \emptyset$ do                         /* the main loop */
4     remove an element w of arg $\preceq$-opt$_{u\in \mathsf{H}}\sigma[u]$ from H
5     foreach x such that ⟨w, x⟩ $\in E$ do
6         $\sigma' \leftarrow \psi(\pi[\mathsf{w}]\hat{\ }\mathsf{x})$
7         if $\sigma' \prec \sigma$[x] then $\sigma$[x] $\leftarrow \sigma'$; $\pi$[x] $\leftarrow \pi[\mathsf{w}]\hat{\ }\mathsf{x}$
```

Notice that Algorithm 1, referred in what follows as **DA**, requires precisely $|V|$-many executions of the main loop, since, after the execution of line 2, nothing is ever inserted again into H. Also, the order of performed operations in the algorithm is not uniquely determined by its structure, since the execution of line 4 may result in choosing different $\preceq$-minimal elements w. This is the reason for the use of phrases "**is guaranteed**" and "**cannot be**" in the theorems that follow.

To state our main theorem, on the correctness of **DA**, we will need the following additional terminology and notation. For $G = \langle V, E\rangle$ and a value-path function $\psi\colon \Pi_G \to [-\infty,\infty]$ define a *max-value path function* $\Psi\colon \Pi_G \to [-\infty,\infty]$ by putting, for every $\langle v_0,\dots,v_\ell\rangle \in \Pi_G$,

$$\Psi(\langle v_0,\dots,v_\ell\rangle) = \max\{\psi(\langle v_0,\dots,v_i\rangle)\colon i=0,1,\dots,\ell\},$$

where maximum is with respect to the order relation $\preceq$. We say that a path $p_v = \langle v_0,\dots,v_\ell\rangle \in \Pi_G$ to $v = v_\ell$:

- is *ψ-optimal* if it is $\preceq$-minimal, that is, provided $\psi(p_v) \preceq \psi(q_v)$ for any other path $q_v \in \Pi_G$ to v;
- is *hereditarily ψ-optimal,* provided $\langle v_0,\dots,v_k\rangle$ is ψ-optimal for every $k \le \ell$;
- is *hereditarily optimal, HO,* provided it is hereditarily ψ-optimal and $\Psi(\langle v_0,\dots,v_k\rangle) \preceq \Psi(p)$ for every hereditarily ψ-optimal p to v_k and all $k \le \ell$;
- is *Ψ-minimal* (in a strong sense) provided $\Psi(p_v) \prec \Psi(q\hat{\ }v)$ for every $q\hat{\ }v \in \Pi_G$ such that $\psi(p_v) \prec \psi(q\hat{\ }v)$ and q is either empty or HO;
- *has the replacement property* when $\psi(\langle v_0,\dots,v_i\rangle) = \psi(q_{v_{i-1}}\hat{\ }v_i)$ for every HO path $q_{v_{i-1}} \in \Pi_G$ to v_{i-1} and all $i \in \{1,\dots,\ell\}$;
- is *monotone* when $\psi(\langle v_0,\dots,v_i\rangle) \preceq \psi(\langle v_0,\dots,v_j\rangle)$ for all $i \le j \le \ell$;
- is *hereditarily ψ-optimal monotone, HOM,* provided it is both hereditarily ψ-optimal and monotone.

Now, we are ready for our main theorem on the correctness of **DA**.

Theorem 1 *Let $G = \langle V, E\rangle$ be a finite directed graph with no loops and $\psi\colon \Pi_G \to [-\infty,\infty]$ be a path-value function. If*

(E) for every $v \in V$ there exists a Ψ-minimal HO path to v with the replacement property,

then the array $\sigma[\,]$ returned by **DA is guaranteed** *to be the ψ-optimal map. Moreover, the array $\pi[\,]$ returned by* **DA** *has the property that, for every $v \in V$, $\pi[v] = \langle v_0,\dots,v_\ell\rangle$ is an HO path to $v = v_\ell$ and $\pi[v_i] = \langle v_0,\dots,v_i\rangle$ every $i \in \{0,\dots,\ell\}$ (i.e., $\{\pi[v]\colon v \in V\}$ is an optimal forest).*

Conversely, if the following monotonicity property holds

(M) $\psi(\langle v_0,\dots,v_i\rangle) \preceq \psi(\langle v_0,\dots,v_\ell\rangle)$ for every path $\langle v_0,\dots,v_\ell\rangle \in \Pi_G$ and $0 \le i < \ell$,

then the function $\sigma[\,]$ returned by **DA cannot be** *ψ-optimal, unless for every $v \in V$ there exists a hereditarily ψ-optimal path to v.*

The proof of Theorem 1 is presented in Sect. 7. In the mean time, we will discuss **DA** and the consequences of Theorem 1.

The two notions involving explicitly the max-value path function Ψ (i.e., HO and Ψ-minimal, used to express (E)) are, at first, hard to fully grasp. Luckily, in most of the applications, they can be replaced by the considerable simpler notion of an HOM path, as it can be seen from the following simple result and the fact that property (M) is satisfied for the vast majority of path-value functions (see, for example, Corollary 2).

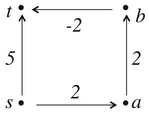

Fig. 1 For this graph, with $S = \{s\}$, indicated weight map ω_E, and path-value function $\psi_{\rm sum}$, (E) is clearly satisfied for every vertex $v \neq t$. It is also satisfied for $v = t$ by a path $p_t = \langle s,a,b,t\rangle$: It is the only $\psi_{\rm sum}$-optimal path, as $\psi_{\rm sum}(p_t) = 2 < 5 = \psi_{\rm sum}(\langle s,t\rangle)$ and also $\Psi_{\rm sum}(p_t) = 4 < 5 = \Psi_{\rm sum}(\langle s,t\rangle)$. However, t does not admit HOM path, as p_t is not monotone

Remark 1 Every HOM path $p = \langle v_0,\dots,v_\ell\rangle \in \Pi_G$ is a Ψ-minimal HO path.

Proof Let $p = \langle v_0,\dots,v_\ell\rangle \in \Pi_G$ be an HOM path and q as in the definition of Ψ-minimality. Then, the monotonicity of p implies that $\Psi(p) = \psi(p)$. Thus, $\Psi(p) = \psi(p) \prec \psi(q\hat{\ }v) \preceq \Psi(q\hat{\ }v)$, that is, p is indeed Ψ-minimal. It is HO since, for every $k \le \ell$ and hereditarily ψ-optimal p to v_k, we have $\Psi(\langle v_0,\dots,v_k\rangle) = \psi(\langle v_0,\dots,v_k\rangle) \preceq \psi(p) \preceq \Psi(p)$. □

Since in the majority of the application of **DA** the Ψ-minimal HO paths that satisfy (E) are actually HOM paths, one might wonder if in all applications of the theorem the phrase "Ψ-minimal HO" can be replaced with "HOM paths." The simplest example that negates such a claim is given by the path-value function $\psi_{\rm last}$ from Example 5: Every path from S is optimal with respect to $\psi_{\rm last}$ (so, (E) is satisfied and, in fact, any spanning forest rooted at S is optimal), while for $V = \{s,c\}$, $E = \{\langle s,c\rangle, \langle c,s\rangle\}$, $S = \{s\}$, and $\omega_V(s) = 1$, $\omega_V(c) = 0$, the vertex c admits no HOM path.

Another example of a path-value function on a graph, which satisfies (E) but has a vertex admitting no HOM path, is presented in Fig. 1. This example uses path-value function $\psi_{\rm sum}$ (Example 1), in which we allow negative values for the edge-weight map ω_E.

From Theorem 1 and Remark 1 it is easy to deduce the following characterization of path-value functions ψ for which **DA** must return the expected optimal map.

Corollary 1 *If $G = \langle V, E\rangle$ and $\psi\colon \Pi_G \to [-\infty,\infty]$ satisfy (M) and the following replacement property*

(R) $\psi(p_{v_\ell}) = \psi(q_{v_{\ell-1}}\hat{\ }v_\ell)$ for every HOM path $p_{v_\ell} = \langle v_0,\dots,v_\ell\rangle$ to v_ℓ and $q_{v_{\ell-1}}$ to $v_{\ell-1}$,

then $\sigma[\,]$ returned by **DA** *is the ψ-optimal map if, and only if, for every $v \in V$ there exists a hereditarily ψ-optimal path to v.*

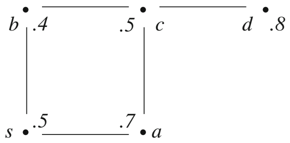

Fig. 2 For the graph, with $S = \{s\}$, neither $p_d = \langle s, a, c, d\rangle$, nor $q_d = \langle s, b, c, d\rangle$ from s to d is hereditarily ψ_{dif}-optimal: Only p_d is optimal, since $\psi_{\text{dif}}(p_d) = .8 - .5 < .8 - .4 = \psi_{\text{dif}}(q_d)$; but the initial segment $\langle s, a, c\rangle$ of p_d is suboptimal, as $\psi_{\text{dif}}(\langle s, a, c\rangle) = .7 - .5 > 5 - .4 = \psi_{\text{dif}}(\langle s, b, c\rangle)$

Proof The existence of hereditarily ψ-optimal paths to every $v \in V$ is a sufficient condition, since, by (M) and (R), every such path is HOM and has the replacement property. So, by Remark 1, (E) holds and Theorem 1 implies that $\sigma[\,]$ returned by **DA** is as needed.

The necessity of the existence of hereditarily ψ-optimal paths follows immediately from the second part of Theorem 1. □

The usefulness of Theorem 1 and Corollary 1 can be appreciated, when noticing how easily one can deduce from them the following two results.

Corollary 2 *The path-value functions ψ_{sum}, $\psi_{\min}$, and ψ_{peak} satisfy the properties (M) and (R). In particular, the* **DA** *algorithm works correctly for these functions.*

Proof The definitions of these functions immediately imply that every path is monotone and that the following strong version (R*) of the replacement property (R) holds:

(R*) $\psi(q_{v_{\ell-1}} \hat{}\, v_\ell) \preceq \psi(p_{v_\ell})$ for all paths $p_{v_\ell} = \langle v_0, \ldots, v_\ell\rangle$ to v_ℓ and $q_{v_{\ell-1}}$ to $v_{\ell-1}$ with $\psi(q_{v_{\ell-1}}) \preceq \psi(p_{v_{\ell-1}})$.

If ψ satisfies (M) and (R*), then every $v \in V$ admits a hereditarily ψ-optimal path to v, see Proposition 2. So, every such path is HOM satisfying the replacement property, and (E) holds. □

At the same time, Theorem 1 easily implies that the reverse is true for the barrier path-value function ψ_{dif} from Example 4. (Compare also [11,38].)

Proposition 1 *The* **DA** *need not return an optimal map, when executed for the path-value function ψ_{dif}.*

Proof For a weighted graph depicted in Fig. 2, which comes from [11], there is no hereditarily ψ_{dif}-optimal path from $S = \{s\}$ to d. Since ψ_{dif} clearly satisfies (M), the result follows from Theorem 1. □

4 Another Variant DA* of Dijkstra's Algorithm

It would have been nice if it had been possible to prove about **DA** that, independently of any extra assumptions on the path-value function ψ,

(•) the family $\{\pi[v] \colon v \in V\}$ returned by the algorithm is always a forest.

Actually, it was claimed in [21, Lemma 2] that **DA** (in their formalism) indeed satisfies (•) (i.e., never returns a cycle) for any path-value function ψ.[1] However, the following simple example shows that a family $\{\pi[v] \colon v \in V\}$ returned by **DA** need not be a forest. The example also shows that the second part of Theorem 1 indeed requires some assumptions on the map ψ.

Example 6 Consider $G = \langle V, E\rangle$ with $V = \{s, a\}$ and $E = \{\langle s, a\rangle, \langle a, s\rangle\}$. Identify $\preceq$ with $\leq$. Define $\psi(p) = 0$ for any path p from s of nonzero length, and $\psi(p) = 1$ for any other path. Then, **DA** returns paths $\pi[s] = \langle s, a, s\rangle$ and $\pi[a] = \langle s, a\rangle$ (as we start with the initialization $\pi[s] = \langle s\rangle$, $\pi[a] = \langle a\rangle$ and, after the first execution of the loop, we have $\pi[s] = \langle s\rangle$, $\pi[a] = \langle s, a\rangle$).

In particular, **DA** returns a non-trivial circular path $\pi[s] = \langle s, a, s\rangle$, which cannot belong to any forest, so (•) is not satisfied. (In the formalism of [21], **DA** returns the predecessor indicators $P[s] = a$ and $P[a] = s$, also a cycle.)

Moreover, **DA** returns $\pi[s] = \psi(\langle s, a, s\rangle) = 0$ and $\pi[a] = \psi(\langle s, a\rangle) = 0$, that is, an optimal map π, in spite of the fact that there is no hereditarily ψ-optimal path in the graph (as any path of length 0 is suboptimal). Thus, the second part of Theorem 1 indeed requires some additional assumptions on ψ.

The property (•) can be ensured by the following simple modification of **DA**, obtained by replacing condition "$\langle \mathsf{w}, \mathsf{x}\rangle \in E$" in line 5 with "$\langle \mathsf{w}, \mathsf{x}\rangle \in E$ and $\mathsf{x} \in \mathsf{H}$." This leads to:

We have the following modification of Theorem 1 for **DA***.

Theorem 2 *Let $G = \langle V, E\rangle$ be a directed graph with no loops and $\psi \colon \Pi_G \to \mathbb{R}$ be a path-value function. If $\pi[\,]$ is returned by* **DA****, then, for every $v \in V$, $\pi[v] = \langle v_0 \ldots, v_\ell\rangle$ is a path with no repeated vertices such that $\pi[v_i] = \langle v_0 \ldots, v_i\rangle$ for every $i \in \{0, \ldots, \ell\}$ (i.e., $\{\pi[v] \colon v \in V\}$*

[1] Apparently, there was a typo in the version of **DA** from [21], since they defined a set $\mathcal{F}$, never used, to avoid reprocessing the vertices in the inner loop of the algorithm. A proper use of $\mathcal{F}$ would make [21, Lemma 2] valid.

Algorithm 2: Dijkstra's algorithm **DA***, aiming to find the ψ-optimal map

```
Data: A finite graph $G = \langle V, E\rangle$ and a path-value function $\psi$
    from $\Pi_G$ to $\langle[-\infty, \infty], \preceq\rangle$
Result: An array $\sigma[\,]$ of numbers, aiming for being $\psi$-optimal
    map
Additional Structure: A variable $\sigma'$, a set H, and an
array $\pi[\,]$ of paths, such that, at any time and for any $v \in V$,
$\pi[v]$ is a path to $v$ with $\sigma[v] = \psi(\pi[v])$
foreach $v \in V$ do $\pi[v] \leftarrow \langle v\rangle$; $\sigma[v] \leftarrow \psi(\pi[v])$
                                    /* initialization loop */
H $\leftarrow V$
while H $\neq \emptyset$ do                        /* the main loop */
    remove an element w of arg $\preceq$-opt$_{u\in H}\sigma[u]$ from H
    foreach x such that $\langle w, x\rangle \in E$ and x $\in$ H do
        $\sigma' \leftarrow \psi(\pi[w]\hat{\ }x)$
        if $\sigma' \prec \sigma[x]$ then $\sigma[x] \leftarrow \sigma'$; $\pi[x] \leftarrow \pi[w]\hat{\ }x$
```

is a forest). If (E) holds, then $\sigma[\,]$ returned by **DA* is guaranteed** *to be the ψ-optimal map. Moreover, the returned map $\pi[\,]$ consists of hereditarily ψ-optimal paths.*

Conversely, if there exists a $v \in V$ such that there is no hereditarily ψ-optimal path to v, then the function $\sigma[\,]$ returned by **DA* cannot be** *ψ-optimal.*

The proof of Theorem 2 is presented in Sect. 7. However, notice that the last part follows immediately from the first part of the theorem, that **DA*** satisfies (•).

Of course, by Theorem 2, if (E) is satisfied, then **DA*** returns an optimum-path forest. But it is worth mentioning that even when the sufficient conditions are not satisfied, the resulting spanning forest from **DA*** (not necessarily optimal) has been useful as an effective image segmentation, see, for example, [1,11,25,27,30,38].

5 Discussion of Properties (E) and "Smooth Function" from [21]

Considering our notation, the properties C1–C3 of path-value functions in [21], called *smooth functions*, can be stated as follows: For any $v_\ell \in V$ there exists a ψ-optimal path $p_{v_\ell} = \langle v_0 \ldots, v_\ell\rangle \in \Pi_G$, with $\ell \geq 0$, such that for $\ell > 0$, if $p_{v_{\ell-1}} = \langle v_0 \ldots, v_{\ell-1}\rangle$, then

C1. $\psi(p_{v_{\ell-1}}) \preceq \psi(p_{v_\ell})$,
C2. $p_{v_{\ell-1}}$ is ψ-optimal, and
C3. $\psi(q_{v_{\ell-1}}\hat{\ }v_\ell) = \psi(p_{v_\ell})$ for any ψ-optimal $q_{v_{\ell-1}} \in \Pi_G$.

The authors claimed in the paper that for any path-value function ψ that satisfies properties C1–C3, **DA** returns the ψ-optimal map $\sigma[\,]$.

The proof of this claim is presented in the appendix of paper [21], where the authors first claim, without a proof, that C1–C3 implies C1*–C3*, which can be stated as follows: For any $v_\ell \in V$ there exists a ψ-optimal path $p_{v_\ell} = \langle v_0 \ldots, v_\ell\rangle \in \Pi_G$, with $\ell \geq 0$, such that for $0 \leq k \leq \ell - 1$ and $\ell > 0$,

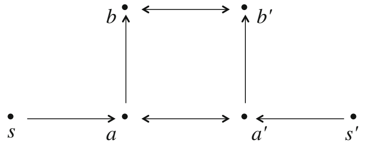

Fig. 3 Graph for Example 8

C1*. $\psi(\langle v_0, \ldots, v_k\rangle) \preceq \psi(p_{v_\ell})$,
C2*. $\langle v_0, \ldots, v_k\rangle$ is ψ-optimal, and
C3*. $\psi(q_{v_k}\hat{\ }\langle v_{k+1}, \ldots, v_\ell\rangle) = \psi(p_{v_\ell})$ for any ψ-optimal path q_{v_k}.

Then, they proceed in proving that for any path-value function satisfying C1*–C3* **DA** must return an optimal mapping.

Unfortunately, neither implication "C1–C3 $\Longrightarrow$ C1*–C3*," nor the claim that conditions C1*–C3* are enough to ensure the optimized output of **DA** is true, as shown by the following three examples.

Example 7 Let $G = \langle V, E\rangle$ be a simple planar grid with $V = \{0, \ldots, 5\} \times \{0, \ldots, 5\}$ considered with 4-adjacency, that is, $\langle(k, \ell), (m, n)\rangle \in E$ precisely when $|k-m|+|\ell-n| = 1$. Let $s_0 = (0, 0)$ and consider standard minimization, that is, $\preceq$ being $\leq$.

For a path $p_{v_\ell} = \langle v_0, \ldots, v_\ell\rangle \in \Pi_G$ in which s_0 appears only at place v_0 we put $\psi(p_{v_\ell}) = \ell$ provided $\ell \leq 3$ and $\psi(p_{v_\ell}) = 0$ otherwise. For a path p_{v_ℓ} in which s_0 appears more than once, or does not appear at all, we put $\psi(p_{v_\ell}) = 100$. Then $\psi(\langle s_0\rangle) = 0$ is optimal. Also, every $v_\ell \in V$ a one-to-one path p_{v_ℓ} of length $\ell \geq 5$, which achieves $\preceq$-minimal value of 0. In addition, the properties C1–C3 are satisfied for any path p_{v_ℓ} of length $\ell \geq 5$. However, only s_0 admits HOM path, so the properties C1*–C2* are not satisfied. Moreover, for any v_1 adjacent to s_0, **DA** returns a suboptimal value 1.

Example 8 Let $G = \langle V, E\rangle$ be as in Fig. 3, that is, with vertices $V = \{s, s', a, a', b, b'\}$ and directed edges $E = \{\langle s, a\rangle, \langle a, b\rangle, \langle s', a'\rangle, \langle a', b'\rangle, \langle a, a'\rangle, \langle a', a\rangle, \langle b, b'\rangle, \langle b', b\rangle\}$. We use the standard minimization (i.e., with $\preceq$ being $\leq$), $S = \{s, s'\}$, and define $\psi(p_v) = 0$ for any $p_v \in \Pi_G$ from S of the following form:

- $v \in \{s, s', a, a'\}$ or having repeated vertices (i.e., not a simple path);
- $\langle \ldots, a', b', b\rangle, \langle s, a, a', b'\rangle, \langle \ldots, a, b, b'\rangle, \langle s', a', a, b\rangle$.

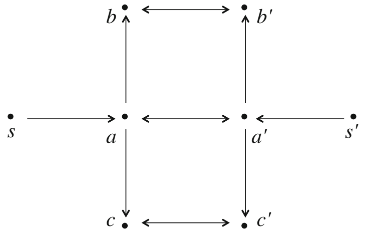

Fig. 4 Graph for Example 9

For all other paths $p_v \in \varPi_G$ we put $\psi(p_v) = 1$.

The path-value function ψ satisfies conditions C1–C3** The path $p_b = \langle s, a, a', b', b\rangle$ satisfies the properties for b: C1* and C2* since it is HOM (as $\psi(\langle s, a, a', b'\rangle) = 0$) and C3*, since the only replacement of $\langle s, a, a', b'\rangle$ in p_b with an optimal path to b' that does not have repeated vertices is p_b itself, while the replacements of any shorter initial segments of p_b are also of the optimal form $\langle \ldots, a', b', b\rangle$. The symmetric argument shows that the path $p_{b'} = \langle s', a', a, b, b'\rangle$ also satisfies conditions C1*–C3*. It is also easy to see that any path from S to $v \in \{b, b', c, c'\}$ satisfies C1*–C3* as well.
DA and DA may terminate with a suboptimal path* Indeed, if the first two vertices removed from H are s and s', then the algorithm will terminate with suboptimal $\pi[b] = \langle s, a, b\rangle$ and $\pi[b'] = \langle s', a', b'\rangle$.
DA and DA may terminate with the optimal map* Indeed, if the first two vertices removed from H are s and a, then the algorithm will terminate with the hereditary optimal $\pi[b] = \langle s, a, a', b', b\rangle$.

Note that the two last claims may happen depending on the tie-breaking policy for removing vertices from H. (In practice, the implementations usually follow the *first-in, first-out* rule, which ensures that s and s' would be removed first.) Apart from that, we provide next an even stronger example, in which the path-value function satisfies C1*–C3* for which the algorithms *cannot* return the optimal map.

Example 9 Let $G' = \langle W, E\rangle$ be as in Fig. 4, and consider the standard minimization (so that $\preceq$ is $\leq$) with $S = \{s, s'\}$ and $\psi(p_v) = 0$ for any $p_v \in \varPi_{G'}$ from S of the form:

- $v \in \{s, s', a, a'\}$ or having repeated vertices (i.e., not a simple path);
- $\langle \ldots, a', b', b\rangle, \langle s, a, a', b'\rangle, \langle \ldots, a, b, b'\rangle, \langle s', a', a, b\rangle$;
- $\langle \ldots, a', c', c\rangle$, $\langle s', a', c'\rangle$, $\langle \ldots, a, c, c'\rangle$, or $\langle s, a, c\rangle$.

For all other paths $p_v \in \varPi_{G'}$ we put $\psi(p_v) = 1$.

The path-value function ψ satisfies conditions C1–C3** The graph G' restricted to the vertices in $V = \{s, s', a, a', b, b'\}$ becomes the graph G from Example 8 with the same path-value function. Since, in G', there are no edges from $\{c, c'\}$ to V, the same paths as in Example 8 show that the conditions C1*–C3* are satisfied for any $v \in V$. The conditions are satisfied for c by a path $p_c = \langle s', a', c', c\rangle$: C1* and C2* since it is HOM and C3*, since the only replacement of $\langle s', a', c'\rangle$ in p_c with an optimal path to c' is $\langle s, a, a', c', c\rangle$, of optimal format $\langle \ldots, a', c', c\rangle$. Similarly, C1*–C3* are satisfied for c' by a path $p_{c'} = \langle s, a, c, c'\rangle$.

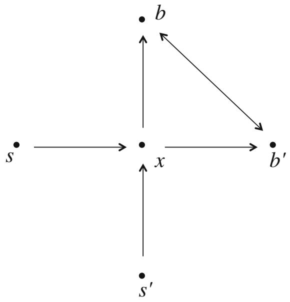

Fig. 5 Graph for Example 10

DA and DA must terminate with a suboptimal path* Indeed, if the algorithm terminates with $\pi[a] = \langle s, a\rangle$ and $\pi[a'] = \langle s', a'\rangle$, then, by Example 8, we end up with suboptimal $\pi[b]$ and $\pi[b']$. Also, termination with $\pi[a] = \langle s', a', a\rangle$ implies that we end up with suboptimal $\pi[c]$, while termination with $\pi[a'] = \langle s, a, a'\rangle$ ensures suboptimality of $\pi[c']$.

The following example shows that the full replacement property we use in Theorem 1 is not necessary for **DA** to work properly. Nevertheless, it is not clear how this assumption could be weakened while keeping the theorems valid.

Example 10 Let G be as in Fig. 5 and use the standard minimization (i.e., with $\preceq$ being $\leq$), $S = \{s, s'\}$ and $\psi(p_v) = 0$ for any $p_v \in \varPi_G$ being one of the two paths $\langle s, x, b, b'\rangle$ and $\langle s', x, b', b\rangle$ or their initial segment. For all other paths $p_v \in \varPi_G$ we put $\psi(p_v) = 1$. Then neither b nor b' admits an optimal path with the replacement property. However, **DA**, as well as **DA***, returns optimal maps: either with $\pi[b] = \langle s', x, b', b\rangle$ or with $\pi[b'] = \langle s, x, b, b'\rangle$.

6 On Optimization of the Algorithms

Remark 2 For the path-value functions ψ satisfying the property (M),[2] the condition "x ∈ H" in line 5 of **DA*** is redundant. On the other hand, under such assumption, it makes sense to keep the condition in line 5 of **DA*** (or even add it to **DA**), since this may reduce an unnecessary computation of $\psi(\pi[\mathrm{w}]\hat{}\mathrm{x})$.

[2] Actually, it is enough to assume only that every hereditarily ψ-optimal path is monotone.

Note also that, for most path-value functions ψ, it is not necessary to keep track of the entire paths $\pi[\mathsf{w}]$ to calculate $\psi(\pi[\mathsf{w}]\hat{}\mathsf{x})$. For example, in the case of the path-value functions ψ_{sum}, $\psi_{\min}$, and ψ_{peak}, from Examples 1–3, we have the equations $\psi_{\text{sum}}(\pi[\mathsf{w}]\hat{}\mathsf{x}) = \sigma[\mathsf{w}] + \omega_E(\mathsf{w},\mathsf{x})$, $\psi_{\min}(\pi[\mathsf{w}]\hat{}\mathsf{x}) = \min\{\sigma[\mathsf{w}], \omega_E(\mathsf{w},\mathsf{x})\}$, and $\psi_{\text{peak}}(\pi[\mathsf{w}]\hat{}\mathsf{x}) = \max\{\sigma[\mathsf{w}], \omega_V(\mathsf{x})\}$, respectively.

Similar simplification is also possible for the barrier distance from Example 4, though in this case, it is necessary to keep record of two functions,

$$\psi^+_{\text{dif}}(\langle v_0,\dots,v_\ell\rangle) = \max_{0\le j\le \ell} \omega_V(v_j)$$

and

$$\psi^-_{\text{dif}}(\langle v_0,\dots,v_\ell\rangle) = \min_{0\le j\le \ell} \omega_V(v_j),$$

updated via $\psi^+_{\text{dif}}(\pi[\mathsf{w}]\hat{}\mathsf{x}) = \max\{\psi^+_{\text{dif}}(\pi[\mathsf{w}]), \omega_V(\mathsf{x})\}$ and $\psi^-_{\text{dif}}(\pi[\mathsf{w}]\hat{}\mathsf{x}) = \min\{\psi^-_{\text{dif}}(\pi[\mathsf{w}]), \omega_V(\mathsf{x})\}$, from which ψ_{dif} is evaluated as $\psi_{\text{dif}}(p_v) = \psi^+_{\text{dif}}(p_v) - \psi^-_{\text{dif}}(p_v)$.

Remark 3 In general, assuming that the path value can be found in $O(1)$-time and that graph degree is of the $O(1)$-order of magnitude, the algorithms can be implemented with H being a binary heap [17] to ensure their termination in $O(n \ln n)$-time, where n is the number of vertices in the graph. This follows from the fact that the main loop is executed precisely n-times and that its execution, finding a vertex with $\preceq$ minimal value of π, takes at most $\ln n$ operations.

The linear time implementation of the algorithms is also possible for sparse graphs and integer path-value functions ψ, as long as the finite initial values $\psi(\langle v_0\rangle)$ and differences $|\psi(p_{v_i}) - \psi(p_{v_{i-1}})|$ for $0 < i \le l$ are less than a number $K > 0$ for any path $p_{v_\ell} \in \Pi_G$ [21]. This requires an efficient bucket sort implementation of H, as the one described in [23].

Remark 4 In some applications, it makes sense for the algorithms to terminate before the main loop is executed for every vertex, giving additional gain in optimization. Examples involve the computation of geodesic paths from a source set to a destination set and shape dilation [21]. In the first case, early termination can occur when a vertex from the destination set is removed from H and, in the second case, when the removed vertex w has optimum-path value $\psi(\pi[\mathsf{w}])$ above a given threshold.

7 Proofs

The following notation and results are for either **DA** and **DA***.

For $k \in \{1,\dots,|V|\}$, let H_k be the state of H immediately before the kth execution of line 5, let w_k be the vertex removed from $\mathsf{H} = \mathsf{H}_k$ during the kth execution of line 5, and let π_k be $\pi[\mathsf{w}_k]$ at that time.[3] First, notice the following lemma that makes no use of the property (E).

Lemma 1 *During the execution of* **DA** *or* **DA*** *and after the initialization loop,*

(i) *for every $v \in V$, $\pi[v]$ is a path to v with $\sigma[v] = \psi(\pi[v])$;*
(ii) *the value of $\sigma[v]$ never increases (in the $\preceq$ sense) and $\pi[v]$ changes only when $\sigma[v]$ decreases;*
(iii) *for every $v \in V$ and $k \in \{0,\dots,|V|\}$, directly after the kth execution of the main loop, either $\pi[v] = \langle v\rangle$ or $\pi[v] = \pi_j\hat{}v$ for some $j \in \{1,\dots,k-1\}$;*
(iv) *for every $\pi_k = \langle v_0,\dots,v_\ell\rangle$ and $i = 1,\dots,\ell-1$, if $v_i = \mathsf{w}_j$, then $\pi_j = \langle v_0,\dots,v_i\rangle$;*
(v) *$\Psi(\pi_i) \preceq \Psi(\pi_j)$ for every $i, j \in \{1,\dots,|V|\}$, $i \le j$.*

Proof (i) Certainly, this holds directly after the initialization loop. Also, the property is preserved when line 7 is executed.
(ii) The values of $\sigma[v]$ or $\pi[v]$ can change only by the execution of line 7 with $\mathsf{x} = v$, when $\sigma[v] = \sigma[\mathsf{x}]$ decreases.
(iii) Certainly, this holds directly after the initialization loop. Also, the property is preserved when line 7 is executed.
(iv) Let $k \in \{1,\dots,|V|\}$. By recursion, it is enough to prove that (iv) holds for this k, as long as it holds for every $k' \in \{1,\dots,k-1\}$. To see that (iv) holds for such a k, notice that, by (iii), either $\pi_k = \langle v\rangle$ or $\pi_k = \pi_i\hat{}v$ for some $i \in \{1,\dots,k-1\}$. Now, if $\pi_k = \langle v\rangle$, then (iv) holds in void, since there is no i for which the condition needs to be checked. On the other hand, if $\pi_{k'} = \pi_i\hat{}v$ for some $k' = i \in \{1,\dots,k-1\}$, then the condition is satisfied by the recursive assumption.
(v) It is enough to prove, by induction on $k \in \{1,\dots,|V|\}$, that the following property

$$I_k\colon\ \Psi(\pi_i) \preceq \Psi(\pi_j) \text{ for every } 1 \le i \le j \le k \text{ holds.}$$

This clearly holds for $k = 1$. So, assume that it holds for some $k < |V|$. We will show that it holds also for $k+1$. For this, it is enough to prove that $\Psi(\pi_k) \preceq \Psi(\pi_{k+1})$.

So, let q be the shortest initial segment of π_k with $\psi(q) = \Psi(\pi_k)$. Then $\psi(q) = \Psi(q)$ and, by (iii), q terminates at w_j for some $j \le k$. In particular, by (iv), $\pi_j = q$ and $\Psi(\pi_j) = \Psi(q) = \psi(q) = \Psi(\pi_k)$. Also, by (iii), π_{k+1} is either $\langle \mathsf{w}_{k+1}\rangle$ or $\pi_i\hat{}\mathsf{w}_{k+1}$ for some $i \in \{1,\dots,k\}$.

First, assume the latter case, that $\pi_{k+1} = \pi_i\hat{}\mathsf{w}_{k+1}$. If $j \le i$, then, by the inductive assumption, $\Psi(\pi_k) = \Psi(\pi_j) \preceq \Psi(\pi_i) \preceq \Psi(\pi_i\hat{}\mathsf{w}_{k+1}) = \Psi(\pi_{k+1})$, as needed.

[3] Notice that in case of the algorithm **DA**, the value of $\pi[\mathsf{w}_k]$ can still further change, as shown in Example 6. But, in the presented argument, π_k remains fixed.

So, assume that $i < j$. Then, after the ith execution of the main loop, we have $\sigma[\mathsf{w}_{k+1}] \preceq \psi(\pi_i\hat{}\mathsf{w}_{k+1})$. Since, by (ii), the value of $\sigma[\mathsf{w}_{k+1}]$ never decreases, the inequality $\sigma[\mathsf{w}_{k+1}] \preceq \psi(\pi_i\hat{}\mathsf{w}_{k+1})$ remains true after the $j-1$th execution of the main loop of the algorithm. In particular, the minimality choice of w_j, ensured in line 4, gives $\psi(\pi_j) = \sigma[\mathsf{w}_j] \preceq \sigma[\mathsf{w}_{k+1}] \preceq \psi(\pi_i\hat{}\mathsf{w}_{k+1})$. Therefore, we have the following inequalities $\Psi(\pi_k) = \Psi(\pi_j) = \psi(\pi_j) \preceq \psi(\pi_i\hat{}\mathsf{w}_{k+1}) \preceq \Psi(\pi_i\hat{}\mathsf{w}_{k+1}) = \Psi(\pi_{k+1})$, as needed.

Finally, assume that $\pi_{k+1} = \langle \mathsf{w}_{k+1}\rangle$. Then, after the initialization, $\sigma[\pi_{k+1}] = \psi(\langle \mathsf{w}_{k+1}\rangle) = \psi(\pi_{k+1})$ and, by (ii), the inequality $\sigma[\mathsf{w}_{k+1}] \preceq \psi(\pi_{k+1})$ remains true after $j-1$th execution of the main loop. So, the minimality choice of w_j, ensured in line 4, gives $\psi(\pi_j) = \sigma[\mathsf{w}_j] \preceq \sigma[\mathsf{w}_{k+1}] \preceq \psi(\pi_{k+1})$. Thus, $\Psi(\pi_k) = \Psi(\pi_j) = \psi(\pi_j) \preceq \psi(\pi_{k+1}) \preceq \Psi(\pi_{k+1})$, as needed. □

The following lemma is the key step in the proof of the theorems.

Lemma 2 *If (E) holds, then after the execution of* **DA** *or* **DA***, *for every* $k \in \{1, \dots, |V|\}$*:*

(P_k) $\pi_k = \pi[\mathsf{w}_k]$ *is HO.*

Proof Choose a $k \in \{1, \dots, |V|\}$ such that (P_j) holds for every $j \in \{1, \dots, k-1\}$. By the power of recursion, it is enough to prove that (P_k) holds as well.

To see (P_k), choose, using (E), a Ψ-minimal HO path $p = \langle v_0, \dots, v_\ell\rangle$ to w_k with the replacement property and notice that it is enough to prove that

$(*)$ $\psi(\pi_k) \preceq \psi(p)$ and $\Psi(\pi_k) \preceq \Psi(p)$.

Indeed, by Lemma 1(iii), $\pi_k = q\hat{}\mathsf{w}_k \in \Pi_G$, where q is either empty or equal to π_j for some $j \in \{1, \dots, k-1\}$. Hence, by the inductive assumption, q is either empty or HO. Thus, the hereditary ψ-optimality of π_k follows from the inequality $\psi(\pi_k) \preceq \psi(p)$, as p is a ψ-optimal path to w_k. Also, hereditary Ψ-optimality of π_k follows from its Ψ-*optimality*, that is, the property that

$\Psi(\pi_k) \preceq \Psi(\pi)$ for every hereditarily ψ-optimal path π to w_k.

But this Ψ-optimality of π_k follows from the inequality $\Psi(\pi_k) \preceq \Psi(p)$, since the HO property of p ensures that $\Psi(p) \preceq \Psi(\pi)$ for every hereditarily ψ-optimal path π to w_k.

To prove $(*)$ first, notice that it holds when $\ell = 0$. Indeed, then we have $p = \langle \mathsf{w}_k\rangle$ and, right after the initialization, $\pi[\mathsf{w}_k]$ is ψ-optimal. Hence, by the parts (i) and (ii) of Lemma 1, the value of $\pi[\mathsf{w}_k]$ remains unchanged during the execution of the algorithm. In particular, $\pi_k = \pi[\mathsf{w}_k] = p = \langle \mathsf{w}_k\rangle$ and $\Psi(\pi_k) = \psi(\pi_k) = \psi(p) = \Psi(p)$, giving us $(*)$. Therefore, in what follows we assume that $\ell > 0$.

Next, notice that $v_\ell = \mathsf{w}_k \in \mathsf{H}_k$. So, there exists the smallest $i \le \ell$ such that $v_i \in \mathsf{H}_k$. Let $t \in \{k, \dots, |V|\}$ be such that $v_i = \mathsf{w}_t$. We will consider several cases.

Case $0 < i = \ell$. Then $v_{\ell-1} \notin \mathsf{H}_k$ and $\pi[v_{\ell-1}] = \pi_j$ for some $j < k$. So, by the inductive assumption, $\pi_j = \pi[v_{\ell-1}]$ is HO. In particular, $\Psi(\pi_j) \preceq \Psi(\langle v_0, \dots, v_{\ell-1}\rangle)$, as $\langle v_0, \dots, v_{\ell-1}\rangle$ is hereditarily ψ-optimal to $v_{\ell-1} = \mathsf{w}_j$. Moreover, by the replacement property, $\psi(\pi_k) = \psi(\pi_j\hat{}\mathsf{w}_k) = \psi(p)$. Thus, we have $\Psi(\pi_k) = \Psi(\pi_j\hat{}\mathsf{w}_k) = \max\{\Psi(\pi_j), \psi(\pi_k)\} \preceq \max\{\Psi(\langle v_0, \dots, v_{\ell-1}\rangle), \psi(p)\} = \Psi(p)$, proving $(*)$. So, in the rest of the argument we will assume that $i < \ell$.

In the rest of the proof we will assume, by way of contradiction, that $(*)$ is false. Notice that this implies that

$(**)$ $\Psi(p) \prec \Psi(\pi_k) = \Psi(q\hat{}\mathsf{w}_k)$, and there exists an $s \in \{1, \dots, k\}$ such that π_s is an initial segment of $\pi_k = q\hat{}\mathsf{w}_k$ for which $\Psi(\pi_s) = \psi(\pi_s) = \Psi(\pi_k)$.

Indeed, if $(*)$ is false, then either $\Psi(p) \prec \Psi(\pi_k)$ or $\psi(p) \prec \psi(\pi_k)$. However, the second of these inequalities implies that $\psi(p) \prec \psi(\pi_k) = \psi(q\hat{}\mathsf{w}_k)$ and, by the Ψ-minimality of p, also $\Psi(p) \prec \Psi(q\hat{}\mathsf{w}_k) = \Psi(\pi_k)$. This shows the first part of $(**)$. To see the second part, notice that if π is the shortest initial segment of π_k for which $\psi(\pi) = \Psi(\pi_k)$ and π is a path to w_s, then s is as needed.

Case $0 = i < \ell$. Then $\langle v_0\rangle$ is ψ-optimal, as an initial segment of HO path p, and so, right after the initialization, $\pi[\mathsf{w}_t] = \pi[v_0] = \langle v_0\rangle$ is ψ-optimal. Hence, by the parts (i) and (ii) of Lemma 1, the value of $\pi[v_0]$ remains unchanged during the execution of the algorithm. In particular, $\pi_t = \langle v_0\rangle$. Moreover, $s \le k \le t$ so that $\mathsf{w}_t = v_0$ belongs to H_s. In particular, the choice of w_s during the sth execution of the algorithm's loop ensures that $\psi(\pi_s) \preceq \psi(\langle v_0\rangle)$. Hence, using $(**)$, $\psi(\pi_s) \preceq \psi(\langle v_0\rangle) \preceq \Psi(p) \prec \Psi(q\hat{}\mathsf{w}_k) = \Psi(\pi_k) = \psi(\pi_s)$, a desired contradiction.

Case $0 < i < \ell$. Then $v_{i-1} \notin \mathsf{H}_k$ and $v_{i-1} = \mathsf{w}_r$ for some $r \in \{1, \dots, k-1\}$. Hence, by the inductive assumption, the path $\pi_r = \pi[\mathsf{w}_r] = \pi[v_{i-1}]$ is HO. Therefore, by the replacement property, we have $\psi(\pi_r\hat{}v_i) = \psi(\langle v_0, \dots, v_i\rangle)$, that is, any time after the rth execution of the main loop the path $\pi_r\hat{}v_i$ to $v_i = \mathsf{w}_t$ is already ψ-optimal. Hence, $\pi_t = \pi[\mathsf{w}_t] = \pi[v_i] = \pi_r\hat{}v_i$.

If $r < s$, then the choice of w_s during the sth execution of the loop ensures that $\psi(\pi_s) \preceq \psi(\pi_t)$ and, using $(**)$, $\psi(\pi_s) \preceq \psi(\pi_t) = \psi(\pi_r\hat{}v_i) = \psi(\langle v_0, \dots, v_i\rangle) \preceq \Psi(p) \prec \Psi(q\hat{}\mathsf{w}_k) = \Psi(\pi_k) = \psi(\pi_s)$, a desired contradiction. So, assume that $r \ge s$. Then, by Lemma 1(v), the property (P_r), and $(**)$, we have $\psi(\pi_s) \preceq \Psi(\pi_s) \preceq \Psi(\pi_r) \preceq$

$\Psi(\langle v_0,\dots,v_r\rangle) \preceq \Psi(p) \prec \Psi(q\hat{}\mathsf{w}_k) = \Psi(\pi_k) = \psi(\pi_s)$, once again a desired contradiction. □

Proof of Theorems 1 and 2 First notice that the family $\{\pi_1,\dots,\pi_{|V|}\}$ forms a forest: This follows from Lemma 1(iv) and the fact that each π_i is a path to different vertex w_i.

The extra condition "$\mathsf{x} \in \mathsf{H}$" in line 5 of **DA*** ensures that the value of $\pi[\mathsf{w}_k]$ does not change from the value of π_k during the jth execution of the loop for every $j \geq k$. Thus, **DA*** returns $\{\pi_1,\dots,\pi_{|V|}\}$ as $\{\pi[v]\colon v \in V\}$, which is a forest, as claimed.

Moreover, when condition (E) holds, Lemma 2 ensures that, after the execution of **DA** or **DA***, the forest $\{\pi_1,\dots,\pi_{|V|}\}$ is optimal. In particular, by Lemma 1(ii), the value of $\pi[\mathsf{w}_k] = \pi_k$ does not change during the jth execution of the loop for every $j \geq k$. So, both algorithms, **DA** as well as **DA***, return $\{\pi_1,\dots,\pi_{|V|}\}$ as $\{\pi[v]\colon v \in V\}$, the optimal forest.

The last part of Theorem 2 follows immediately from its first part, that for path returned **DA*** among $P = \{\pi[v]\colon v \in V\}$, all its initial segments must be also in P, so that, if all paths in P are ψ-optimal, then they must be also hereditarily ψ-optimal.

Similarly, to prove the last part of Theorem 1 it is enough to show that, under the assumption of (M), the family $P = \{\pi[v]\colon v \in V\}$ returned by **DA** coincides with $\{\pi_1,\dots,\pi_{|V|}\}$, which is a forest. But this immediately follows from Lemma 3 below. □

Lemma 3 **DA** *executed with a path-value function satisfying (M) returns the map* $\pi[\,]$ *such that for every* $v_\ell \in V$*, if* $\pi[v_\ell] = \langle v_0 \dots, v_\ell\rangle$*, then* $\pi[v_i] = \langle v_0 \dots, v_i\rangle$ *for every* $i \in \{0,\dots,\ell\}$.

Proof Since the algorithm terminates with H empty, it is enough to prove that the following properties hold any time after the initialization loop.

(i) For every $v_\ell \in V\backslash\mathsf{H}$, if $\pi[v_\ell] = \langle v_0 \dots, v_\ell\rangle$, then, for every $i \in \{0,\dots,\ell\}$, $v_i \in V\backslash\mathsf{H}$ and $\pi[v_i] = \langle v_0 \dots, v_i\rangle$.

(ii) $\psi(\pi[\mathsf{w}_i]) \preceq \psi(\pi[\mathsf{w}_j]) \preceq \psi(\pi[\mathsf{u}])$ for every $\mathsf{u} \in \mathsf{H}$ and $\mathsf{w}_i, \mathsf{w}_j \in V\backslash\mathsf{H}$ with $i \leq j$.

Certainly, this holds directly after the initialization loop. Thus, it is enough to show that (i) and (ii) are preserved by any, say kth, execution of the main loop.

Indeed, the properties (i) and (ii) are preserved when we remove w from H by the execution of line 4. For (ii), this follows from the $\preceq$-minimality imposed on $\mathsf{w} = \mathsf{w}_k$ removed from H. To see (i), notice that, by Lemma 1(iii), just before execution of line 4, $\pi[\mathsf{w}]$ equals either $\langle \mathsf{w}\rangle$ or $\pi[\mathsf{u}]\hat{}\mathsf{w}$, where $\mathsf{u} \in V\backslash\mathsf{H}$ and $\pi[\mathsf{u}]$ satisfies (i). In either case, $\pi[\mathsf{w}]$ satisfies (i), after w is removed from H.

So, it is enough to show that each execution of line 7 preserves (i) and (ii). Indeed, we can execute the commands $\sigma[\mathsf{x}] \leftarrow \sigma'$ and $\pi[\mathsf{x}] \leftarrow \pi[\mathsf{w}]\hat{}\mathsf{x}$ only when $\mathsf{x} \in \mathsf{H}$ since, by (M), any $\mathsf{x} \in \mathsf{V}\backslash\mathsf{H}$ is equal to w_i for some $i \in \{1,\dots,k\}$ and $\sigma' = \psi(\pi[\mathsf{w}]\hat{}\mathsf{x}) \succeq \psi(\pi[\mathsf{w}]) = \psi(\pi[\mathsf{w}_k]) \succeq \psi(\pi[\mathsf{w}_i]) = \psi(\pi[\mathsf{x}]) = \sigma[\mathsf{x}]$, where the first inequality is justified by (M) and the second by (ii). Thus, the condition $\sigma' \prec \sigma[\mathsf{x}]$ in line 7 is not satisfied, so the rest of the line is not executed. Hence, it is enough to show that execution of line 7 with $\mathsf{x} \in \mathsf{H}$ preserves (i) and (ii).

In this case, after line 7 is executed, we still have $\psi(\pi[\mathsf{w}_i]) \preceq \psi(\pi[\mathsf{w}_k]) = \psi(\pi[\mathsf{w}]) \preceq \psi(\pi[\mathsf{w}]\hat{}\mathsf{x}) = \psi(\pi[\mathsf{x}])$, preserving (ii). At the same time, (i) cannot be affected by a change of $\pi[\mathsf{x}]$ when $\mathsf{x} \in \mathsf{H}$. This completes the proof of Theorems 1 and 2. □

The only remaining proof we still need is that of

Proposition 2 *If* ψ *satisfies (M) and*

(R)* $\psi(q_{v_{\ell-1}}\hat{}v_\ell) \preceq \psi(p_{v_\ell})$ *for all paths* $p_{v_\ell} = \langle v_0,\dots,v_\ell\rangle$ *to* v_ℓ *and* $q_{v_{\ell-1}}$ *to* $v_{\ell-1}$ *with* $\psi(q_{v_{\ell-1}}) \preceq \psi(p_{v_{\ell-1}})$,

then every $v \in V$ *admits a hereditarily* ψ*-optimal path to* v.[4]

Proof First notice that, by the properties (M) and (R*), for every path $\langle v_0,\dots,v_\ell\rangle$, if $v_i = v_j$ for some $i \leq j \leq \ell$, then we have $\psi(\langle v_0,\dots,v_i\rangle) \preceq \psi(\langle v_0,\dots,v_j\rangle)$ and $\psi(\langle v_0,\dots,v_i,v_{j+1},\dots,v_\ell\rangle) \preceq \psi(\langle v_0,\dots,v_\ell\rangle)$. Thus, for every path p_v to v there exists a path q_v to v which contains no repeated vertices and such that $\psi(q_v) \preceq \psi(p_v)$. In particular, for every $v \in V$ there exists a number $\psi(v)$, the strength of the ψ-optimal path to v: It is the $\preceq$-smallest among the numbers $\psi(q_v)$, where q_v is a path to v with no repeated vertices.

Now, suppose the proposition is false. Among $v \in V$ for which no path to v is hereditarily ψ-optimal, pick a point v^* for which $\psi(v^*)$ is $\preceq$-minimal and let $p_{v_\ell} = \langle v_0,\dots,v_\ell\rangle$ be a path to v^* with $\psi(p_{v_\ell}) = \psi(v^*)$. Choose the greatest index $k \in \{0,\dots\ell-1\}$ for which the inequality $\psi(v_k) \preceq \psi(\langle v_0,\dots,v_k\rangle)$ is not the equation. It exists since p_{v_ℓ} cannot be hereditarily ψ-optimal. Therefore, $\psi(v_k) \prec \psi(\langle v_0,\dots,v_k\rangle) \preceq \psi(p_{v_\ell}) = \psi(v^*)$, so, by the $\preceq$-minimality of $\psi(v^*)$, there exists a hereditarily ψ-optimal path q_{v_k} to v_k. Now, by induction on $n \in \{k,\dots,\ell\}$, we prove that there exists a hereditarily ψ-optimal path q_{v_n} to v_n. Clearly, it is true for $n = k$. Also, if it is true for some $n \in \{k,\dots,\ell-1\}$, then it is also true for $n+1$.

[4] Note that if we weaken the assumptions by replacing (R*) with the property (R$^+$) obtained by replacing in (R*) symbols $\preceq$ with the equation $=$, then the implication does not hold any more: ψ_{dif} from Example 4 satisfies (M) and, for the example from Fig. 2, also (R$^+$), but fails the conclusion of Proposition 2.

Indeed, as $\psi(q_{v_n}) = \psi(v_n) \preceq \psi(\langle v_0, \ldots, v_n \rangle)$, (R*) implies $\psi(v_{n+1}) \preceq \psi(q_{v_n} \hat{} v_{n+1}) \preceq \psi(\langle v_0, \ldots, v_{n+1} \rangle) = \psi(v_{n+1})$, where the last equation follows from the definition of k. Thus, $\psi(q_{v_n} \hat{} v_{n+1}) = \psi(v_{n+1})$ and, since q_{v_n} is hereditarily ψ-optimal, so is $q_{v_{n+1}} = q_{v_n} \hat{} v_{n+1}$, finishing the induction.

Now, q_{v_ℓ} is a hereditarily ψ-optimal path to $v_\ell = v^*$, contradicting the choice of v^*. □

8 Conclusion

We presented the conditions of path-value functions on directed graphs that ensure the correct behavior of the Dijkstra-type algorithms and discussed the benefits of such result to image processing. This result, with the proposed graph-search algorithm **DA***, can be used to guide the design of new operators based on the image foresting transform, IFT. As future work, we intend to present a survey of IFT-based operators for image processing and analysis.

A recent work [34] has appeared as a survey on the all-pairs shortest path problem for the case of the additive path-value function (Example 1). Therefore, the present work also creates opportunity for further investigation of the all-pairs shortest path problem for path-value functions that satisfy condition (E), as well as of solutions to new problems in other applications of Dijkstra's algorithm.

References

1. Alexandre, E.B., Chowdhury, A.S., Falcão, A.X., de Miranda, P.A.V.: IFT-SLIC: a general framework for superpixel generation based on simple linear iterative clustering and image foresting transform. In: XXVIII SIBGRAPI—Conference on Graphics, Patterns and Images, pp. 337–344 (2015)
2. Amorim, W.P., Falcão, A.X., Papa, J.P., Carvalho, M.H.: Improving semi-supervised learning through optimum connectivity. Pattern Recognit. **60**, 72–85 (2016)
3. Andaló, F.A., Miranda, P.A.V., Torres, R.da S., Falcão, A.X.: Shape feature extraction and description based on tensor scale. Pattern Recognit. **43**(1), 26–36 (2010)
4. Bai, X., Sapiro, G.: Geodesic matting: a framework for fast interactive image and video segmentation and matting. Int. J. Comput. Vision **82**, 113–132 (2009)
5. Bergo, F.P.G., Falcão, A.X., Miranda, P.A.V., Rocha, L.M.: Automatic image segmentation by tree pruning. J. Math. Imaging Vision **29**(2–3), 141–162 (2007)
6. Cappabianco, F.A.M., Falcão, A.X., Yasuda, C.L., Udupa, J.K.: Brain tissue MR-image segmentation via optimum-path forest clustering. Comput. Vision Image Underst. **116**(10), 1047–1059 (2012)
7. Carvalho, B.M., Gau, C.J., Herman, G.T., Kong, T.Y.: Algorithms for fuzzy segmentation. Pattern Anal. Appl. **2**, 73–81 (1999)
8. Carvalho, B.M., Herman, G.T., Kong, T.Y.: Simultaneous fuzzy segmentation of multiple objects. Discrete Appl. Math. **151**, 55–77 (2005)
9. Ciesielski, K.C., Herman, G.T., Kong, T.Y.: General theory of fuzzy connectedness segmentations. J. Math. Imaging Vision **55**(3), 304–342 (2016)
10. Ciesielski, K.C., Miranda, P.A.V., Falcão, A.X., Udupa, J.K.: Joint graph cut and relative fuzzy connectedness image segmentation algorithm. Med. Image Anal. **17**, 1046–1057 (2013)
11. Ciesielski, K.C., Strand, R., Malmberg, F., Saha, P.K.: Efficient algorithm for finding the exact minimum barrier distance. Comput. Vision Image Underst. **123**, 53–64 (2014)
12. Ciesielski, K.C., Udupa, J.K.: Affinity functions in fuzzy connectedness based image segmentation I: equivalence of affinities. Comput. Vision Image Underst. **114**, 146–154 (2010)
13. Ciesielski, K.C., Udupa, J.K.: Affinity functions in fuzzy connectedness based image segmentation II: defining and recognizing truly novel affinities. Comput. Vision Image Underst. **114**, 155–166 (2010)
14. Ciesielski, K.C., Udupa, J.K.: Region-based segmentation: fuzzy connectedness, graph cut, and other related algorithms. In: Deserno, T.M. (ed.) Biomedical Image Processing, pp. 251–278. Springer, Berlin (2011)
15. Ciesielski, K.C., Udupa, J.K., Falcão, A.X., Miranda, P.A.V.: Fuzzy connectedness image segmentation in graph cut formulation: a linear-time algorithm and a comparative analysis. J. Math. Imaging Vision **44**, 375–398 (2012)
16. Ciesielski, K.C., Udupa, J.K., Saha, P.K., Zhuge, Y.: Iterative relative fuzzy connectedness for multiple objects, allowing multiple seeds. Comput. Vision Image Underst. **107**, 160–182 (2007)
17. Cormen, T., Leiserson, C., Rivest, R.: Introduction to Algorithms. MIT Press, Cambridge (1990)
18. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische Mathematik **1**, 269–271 (1959)
19. Falcão, A.X., Costa, L.F., da Cunha, B.S.: Multiscale skeletons by image foresting transform and its applications to neuromorphometry. Pattern Recognit. **35**(7), 1569–1580 (2002)
20. Falcão, A.X., da Cunha, B.S., Lotufo, R.A.: Design of connected operators using the image foresting transform. SPIE Med. Imaging **4322**, 468–479 (2001)
21. Falcão, A.X., Stolfi, J., Lotufo, R.A.: The image foresting transform: theory, algorithms, and applications. IEEE Trans. Pattern Anal. Mach. Intell. **26**, 19–29 (2004)
22. Falcão, A.X., Udupa, J.K.: A 3D generalization of user-steered live wire segmentation. Med. Image Anal. **4**(4), 389–402 (2000)
23. Falcão, A.X., Udupa, J.K., Miyazawa, F.K.: An ultra-fast user-steered image segmentation paradigm: live-wire-on-the-fly. IEEE Trans. Med. Imaging **19**(1), 55–62 (2000)
24. Frieze, A.: Minimum paths in directed graphs. Oper. Res. Q. **28**(2), 339–346 (1977)
25. Herman, G.T., Carvalho, B.M.: Multiseeded segmentation using fuzzy connectedness. IEEE Trans. Pattern Anal. Mach. Intell. **23**, 460–474 (2001)
26. Lotufo, R.A., Falcão, A.X., Zampirolli, F.A.: IFT-watershed from gray scale marker. In: XV SIBGRAPI—Brazillian Symp. on Computer Graphics and Image Processing, pp. 146–152 (2002)
27. Mansilha, L.A.C., Jackowski, M.P., Miranda, P.A.V.: Image foresting transform with geodesic star convexity for interactive image segmentation. In: 20th IEEE Conference on Image Processing (ICIP), pp. 4054–4058 (2013)
28. Mansilla, L.A.C., Miranda, P.A.V. Cappabianco, F.A.M.: Oriented image foresting transform segmentation with connectivity constraints. In: IEEE International Conference on Image Processing (ICIP), pp. 289–296 (2016)
29. Miranda, P.A.V., Falcão, A.X., Spina, T.V.: Riverbed: a novel user-steered image segmentation method based on optimum boundary tracking. IEEE Trans. Image Process. **21**(6), 3042–3052 (2012)
30. Miranda, P.A.V., Mansilla, L.A.C.: Oriented image foresting transform segmentation by seed competition. IEEE Trans. Image Process. **23**(1), 389–398 (2014)

31. Papa, J.P., Falcão, A.X., de Albuquerque, V.H.C., Tavares, J.M.R.S.: Efficient supervised optimum-path forest classification for large datasets. Pattern Recognit. **45**(1), 512–520 (2012)
32. Papa, J.P., Falcão, A.X., Suzuki, C.T.N.: Supervised pattern classification based on optimum-path forest. Int. J. Imaging Syst. Technol. **19**(2), 120–131 (2009)
33. Ragnemalm, I.: Fast erosion and dilation by contour processing and thresholding of distance maps. Pattern Recognit. Lett. **13**, 161–166 (1992)
34. Reddy, K.R.U.K.: A survey of the all-pairs shortest paths problem and its variants in graphs. Acta Univ. Sapientiae Informatica **8**(1), 16–40 (2016)
35. Rocha, L.M., Cappabianco, F.A.M., Falcão, A.X.: Data clustering as an optimum-path forest problem with applications in image analysis. Int. J. Imaging Syst. Technol. **19**(2), 50–68 (2009)
36. Saito, P.T.M., Suzuki, C.T.N., Gomes, J.F., de Rezende, P.J., Falcão, A.X.: Robust active learning for the diagnosis of parasites. Pattern Recognit. **8**(11), 3572–3583 (2015)
37. Spina, T.V., de Miranda, P.A.V., Falcão, A.X.: Hybrid approaches for interactive image segmentation using the Live Markers paradigm. IEEE Trans. Image Process. **23**(12), 5756–5769 (2014)
38. Strand, R., Ciesielski, K.C., Malmberg, F., Saha, P.K.: The minimum barrier distance. Comput. Vision Image Underst. **117**(4), 429–437 (2013)
39. Torres, R.S., Falcão, A.X.: Contour salience descriptors for effective image retrieval and analysis. Image Vision Comput. **25**(1), 3–13 (2007)
40. Torres, R.S., Falcão, A.X., Costa, L.F.: A graph-based approach for multiscale shape analysis. Pattern Recognit. **37**(6), 1163–1174 (2004)
41. Udupa, J.K., Saha, P.K., Lotufo, R.A.: Relative fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **24**, 1485–1500 (2002)
42. Udupa, J.K., Samarasekera, S.: Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. Graph. Models Image Process. **58**, 246–261 (1996)

Krzysztof Chris Ciesielski received his Master and Ph.D. degrees in pure mathematics from Warsaw University, Poland, in 1981 and 1985, respectively. He works at West Virginia University since 1989. In addition, since 2006 he holds a position of Adjunct Professor in the Department of Radiology at the University Pennsylvania. He is an author or coauthor of three books and over 130 journal research papers. His research interests include both pure mathematics (real analysis, topology, set theory) and applied mathematics (image processing, especially image segmentation).

Alexandre Xavier Falcão is full-time professor at the Institute of Computing, University of Campinas, Campinas, SP, Brazil. He received a B.Sc. in Electrical Engineering from the Federal University of Pernambuco, Recife, PE, Brazil, in 1988. He has worked in biomedical image processing, visualization, and analysis since 1991. In 1993, he received an M.Sc. in Electrical Engineering from the University of Campinas, Campinas, SP, Brazil. During 1994–1996, he worked with the Medical Image Processing Group at the Department of Radiology, University of Pennsylvania, PA, the USA, on interactive image segmentation for his doctorate. He got his doctorate in Electrical Engineering from the University of Campinas in 1996. In 1997, he worked in a project for Globo TV at a research center, CPqD-TELEBRAS in Campinas, developing methods for video quality assessment. His experience as professor of Computer Science and Engineering started in 1998 at the University of Campinas. His main research interests include image/video processing, visualization, and analysis; graph algorithms and dynamic programming; image annotation, organization, and retrieval; machine learning and pattern recognition; and image analysis applications in biology, medicine, biometrics, geology, and agriculture.

Paulo A. V. Miranda is currently professor at the Institute of Mathematics and Statistics (IME) of the University of São Paulo (USP), SP, Brazil. He received a B.Sc. in Computer Engineering (2003) and an M.Sc. in Computer Science (2006) from the University of Campinas (UNICAMP), SP, Brazil. In 2008, he received the best Master's Dissertation Award from the Institute of Computing (UNICAMP). During 2008–2009, he was with the Medical Image Processing Group, Department of Radiology, University of Pennsylvania, Philadelphia, the USA, where he worked on image segmentation for his doctorate. He got his doctorate in Computer Science from the University of Campinas (UNICAMP) in 2009. After that, he worked as a post-doctoral researcher in a project in conjunction with the professors of the Department of Neurology, Unicamp. His experience as professor of Computer Science and Engineering started in 2011 at the University of São Paulo. His main research involves image segmentation and analysis, medical imaging applications, pattern recognition, and content-based image retrieval.