



# Analysis of Artifacts in Shell-Based Image Inpainting: Why They Occur and How to Eliminate Them

L. Robert Hocking<sup>1</sup> · Thomas Holding<sup>2</sup> · Carola-Bibiane Schönlieb<sup>1</sup>

Received: 11 December 2018 / Revised: 19 December 2019 / Accepted: 19 December 2019 /

Published online: 16 March 2020

© The Author(s) 2020

## Abstract

In this paper we study a class of fast geometric image inpainting methods based on the idea of filling the inpainting domain in successive shells from its boundary inwards. Image pixels are filled by assigning them a color equal to a weighted average of their already filled neighbors. However, there is flexibility in terms of the order in which pixels are filled, the weights used for averaging, and the neighborhood that is averaged over. Varying these degrees of freedom leads to different algorithms, and indeed the literature contains several methods falling into this general class. All of them are very fast, but at the same time all of them leave undesirable artifacts such as “kinking” (bending) or blurring of extrapolated isophotes. Our objective in this paper is to build a theoretical model in order to understand why these artifacts occur and what, if anything, can be done about them. Our model is based on two distinct limits: a continuum limit in which the pixel width  $h \rightarrow 0$  and an asymptotic limit in which  $h > 0$  but  $h \ll 1$ . The former will allow us to explain “kinking” artifacts (and what to do about them) while the latter will allow us to understand blur. Both limits are derived based on a connection between the class of algorithms under consideration and stopped random walks. At the same time, we consider a semi-implicit extension in which pixels in a given shell are solved for simultaneously by solving a linear system. We prove (within the continuum limit) that this extension is able to completely eliminate kinking artifacts, which we also prove must always be present in the direct method. Finally, we show that although our results are derived in the context of inpainting, they are

---

Communicated by Hans Munthe-kaas.

---

CBS acknowledges support from the Leverhulme Trust project ‘Breaking the non-convexity barrier’, the EPSRC Grant ‘EP/M00483X/1’, EPSRC centre ‘EP/N014588/1’, the Cantab Capital Institute for the Mathematics of Information, the CHiPS (Horizon 2020 RISE Project Grant), the Global Alliance Project ‘Statistical and Mathematical Theory of Imaging’ and the Alan Turing Institute. RH and TH acknowledge support from the EPSRC Cambridge Centre for Analysis and the Cambridge Trust.

---

✉ L. Robert Hocking  
rob.l.hocking@gmail.com

Extended author information available on the last page of the article

in fact abstract results that apply more generally. As an example, we show how our theory can also be applied to a problem in numerical linear algebra.

**Keywords** Image processing · Image inpainting · Partial differential equations · Stopped random walks · Numerical analysis

**Mathematics Subject Classification** 68U10 · 65M12 · 65M15 · 65F10 · 60G50 · 60G40 · 35F15 · 60G42 · 65M15 · 35Q68

## 1 Introduction

Image inpainting refers to the filling in of a region in an image where information is missing or needs to be replaced (due to, for example, an occluding object), in such a way that the result looks plausible to the human eye. The region to be filled is subsequently referred to as the inpainting domain. Since the seminal work of Bertalmio et al. [6], image inpainting has become increasingly important, with applications ranging from removing an undesirable or occluding object from a photograph, to painting out a wire in an action sequence, to 3D conversion of film [27], as well as 3D TV and novel viewpoint construction [12,17,29] in more recent years. See [21] for a recent survey of the field.

Image inpainting methods can loosely be categorized as *exemplar-based* and *geometric*. Exemplar-based methods generally operate by copying pieces of the undamaged portion of the image into the inpainting domain, in such a way as to make the result appear seamless. Examples include [3,16,43]. Exemplar-based methods also operate behind the scenes in Photoshop’s famous *Content-Aware Fill* tool. These methods excel at interpolating texture across large gaps, but may produce artifacts in structure dominated images with strong edges, or be needlessly expensive if the inpainting domain is thin.

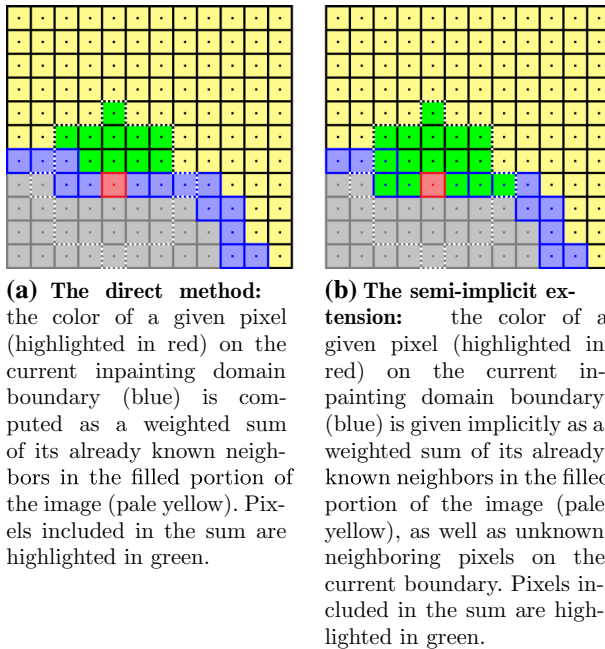
Geometric inpainting methods attempt to smoothly extend image structure into the inpainting domain, typically using partial differential equations (PDEs) or variational principles—see [34] for a comprehensive survey. Examples based on PDEs include the seminal work of Bertalmio et al. [6], methods based on anisotropic diffusion such as [40], while examples based on variational principles include TV,  $TV-H^{-1}$ , Mumford–Shah, Cahn–Hilliard inpainting [11,14], Euler’s Elastica [13,33], as well as the joint interpolation of image values and a guiding vector field in Ballester et al. [4]. These approaches are typically iterative and convergence is often slow (due to either the high order of the underlying PDE model or the difficult nature of the variational problem). On the other hand, Telea’s Algorithm [38], coherence transport [7,31], and our previous work Guidefill [27] are based on the simple idea of filling the inpainting domain in successive shells from the boundary inward, setting the color of pixels on the current boundary equal to a weighted average of their already filled neighbors—Fig. 1 illustrates this process. In general, “neighbors” may include “ghost pixels” lying between pixel centers and defined using bilinear interpolation—this concept was introduced in [27]. By choosing the weights, fill order, and averaging neighborhood carefully, image isophotes may be extrapolated into the inpainting domain. Compared



**Fig. 1** Shell-based inpainting: here we illustrate the shell-based inpainting of an image including an undesirable human to be removed. In **a** we see the original image, including a human that is gradually eroded in **b–d** as we fill more shells. In this case the inpainting method is Guidefill [27], and the application is disocclusion for 3D conversion, which means that the human does not need to be removed entirely. See [27] or [34, Chapter 9] for more details on this application

to their iterative counterparts, these methods have the advantage of being very fast. However, at the same time all of them create, to a greater or lesser degree, some potentially disturbing visual artifacts. For example, extrapolated isophotes may “kink” (change direction), blur, or end abruptly. Although generally these problems have been gradually reduced over time as newer and better algorithms have been proposed, they are still present to some degree even in the state of the art.

The present paper has two goals. Firstly, to analyze these algorithms as a class in order to understand the origins of these artifacts and what, if anything, can be done about them. Secondly, to propose a simple extension in which the pixels in a given shell are solved simultaneously as a linear system, and to analyze its benefits. We will refer to the original methods as “direct” and the extended methods as “semi-implicit”—see Fig. 2 for an illustration of the difference between them. To our knowledge, this extension has never before been proposed in the literature. We also propose a simple iterative method for solving the linear system arising in the semi-implicit extension, show that it is equivalent to damped-Jacobi or successive over-relaxation (SOR), and analyze its convergence rates when applied to the linear system arising in semi-implicit Guidefill. Successive over-relaxation is shown to converge extremely quickly, provided unknowns are ordered appropriately.



**Fig. 2** The direct method and its semi-implicit extension: in this illustration, the filled portion of the image is highlighted in pale yellow, and the current inpainting domain is highlighted in both grey and blue, with the former denoting pixels in the interior of the inpainting domain and the latter pixels on its current boundary. The boundary is the “shell” that is currently being filled—note that the filled (known) portion of the image consists not only of the original undamaged region, but also of previously filled shells. At this stage, the grey and blue pixels are both unknown. In the direct method (a), the color of a given pixel (highlighted in red) is computed directly as a weighted average of pixels (in green) within a given neighborhood (outlined in white) that are already known. In the semi-implicit extension, the sum also includes pixels on the current boundary of the inpainting domain, but not pixels in its interior. This results in linear system that needs to be solved, but this can be done relatively cheaply (see Sect. 6.2) and has benefits in terms of artifact reduction (see Sect. 7.1)

**Remark 1** The original motivation for this paper comes from one of the author’s experiences working in industry for the 3D conversion company Gener8 (see [27] or [34, Chapter 9] for a description of 3D conversion). The programmers there had implemented a series of shell-based inpainting methods of the form discussed above (all essentially variants of Telea’s algorithm [38], which they were unaware of) in order to solve an inpainting problem arising in their 3D conversion pipeline. This type of algorithm was attractive to them because it is fast and simple, while generally adequate for their application (where inpainting domains are typically very thin). However, they were puzzled by some of the artifacts they were seeing, most notably kinking artifacts. After a literature review it became clear that the existing theory [7,31,32] was enough to explain some, but not all, of what they observed. This paper is an attempt to fill the gap.

## 1.1 Organization

The remainder of this paper is organized as follows. First, in Sect. 2, we present the general form of the class of inpainting methods under consideration, including pseudocode for both the direct method and an implementation of our proposed semi-implicit extension (see Algorithm 1). We also cover in detail known artifacts created by the direct form of the method, how they have been reduced over time, related work and our contributions. Next, in Sect. 3 we review the main variants of Algorithm 1 present in the literature. Section 4 describes an equivalence principle between weighted sums of ghost pixels and a sum over real pixels with different weights. This principle will be applied repeatedly throughout our theoretical analysis. Section 5 describes our semi-implicit extension of Algorithm 1, including a description of an implementation of two alternatives for solving the resulting linear system, namely damped Jacobi and successive over-relaxation (SOR). Section 6 contains our core analysis, and is divided into a number of subsections. The first of these, Sect. 6.1, describes the simplifying assumptions that we make throughout. Next, Sect. 6.2 presents a theoretical analysis of the convergence rates of damped Jacobi and SOR for solving the linear system arising in semi-implicit Guidefill, and compares with real experiments. In Sect. 6.3 we define the continuum and asymptotic limits of Algorithm 1. Section 6.4 describes the connection between Algorithm 1 and stopped random walks, while Sects. 6.5 and 6.7 use this connection to prove convergence to the asymptotic and continuum limits previously defined in Sect. 6.3. Next, In Sect. 7, we apply our analysis to gain a theoretical understanding of kinking and blur artifacts. In particular, Sect. 7.1 utilizes our continuum limit to both explain kinking artifacts, and to show how to overcome them. Section 7.3 utilizes the asymptotic limit to make theoretical predictions quantifying the extent to which a given method introduces blur artifacts. Sections 7.1 and 7.3 also contain numerous numerical experiments demonstrating the accuracy of the predictions made by these limits. In Sect. 8 we show that our results—although derived in the context of image inpainting—are abstract results that can be applied to other problems. To demonstrate this, we apply our asymptotic limit to a problem in numerical linear algebra. Finally in Sect. 9 we draw some conclusions and sketch some directions for potential future research. We also have a number of appendices giving technical details of certain proofs.

## 1.2 Notation

- $h$  = the width of one pixel.
- $\mathbb{Z}_h^2 := \{(nh, mh) : (n, m) \in \mathbb{Z}^2\}$ .
- Given  $\mathbf{v} \in \mathbb{R}^2$ , we denote by  $L_{\mathbf{v}} := \{\lambda \mathbf{v} : \lambda \in \mathbb{R}\}$  the line through the origin parallel to  $\mathbf{v}$ .
- Given  $\mathbf{x} \in \mathbb{R}^2$ , we denote by  $\theta(\mathbf{x}) \in [0, \pi)$  the counter-clockwise angle  $L_{\mathbf{x}}$  makes with the  $x$ -axis.  $\theta(\mathbf{x})$  can also be thought of as the counterclockwise angle  $\mathbf{x}$  makes with the  $x$ -axis, modulo  $\pi$ .
- Given  $\mathbf{v} \in \mathbb{R}^2$ , we denote by  $\mathbf{v}^\perp$  the counterclockwise rotation of  $\mathbf{v}$  by  $90^\circ$ .

- $\Omega = [a, b] \times [c, d]$  and  $\Omega_h = \Omega \cap \mathbb{Z}_h^2$  are the continuous and discrete image domains.
- $D_h = D_h^{(0)} \subset \Omega_h$  is the (initial) discrete inpainting domain.
- $D_h^{(k)} \subseteq D_h^{(0)}$  is the discrete inpainting domain on step  $k$  of the algorithm.
- $D \subset \Omega := \{\mathbf{x} \in \Omega : \exists \mathbf{y} \in D_h \text{ s.t. } \|\mathbf{y} - \mathbf{x}\|_\infty < h\}$  is the continuous inpainting domain.
- $D^{(k)}$  is the continuous inpainting domain on step  $k$  of the algorithm, defined in the same way as  $D$ .
- $u_0 : \Omega_h \setminus D_h \rightarrow \mathbb{R}^d$  is the given (discrete) image. In an abuse of notation, we also use  $u_0$  to denote an assumed underlying continuous image  $u_0 : \Omega \setminus D \rightarrow \mathbb{R}^d$ .
- $u_h : \Omega_h \rightarrow \mathbb{R}^d$  is the inpainted completion of  $u_0$ .
- $\mathbf{g}(\mathbf{x})$  is the guidance vector field used by coherence transport and Guidefill.
- $B_\epsilon(\mathbf{x})$  the solid ball of radius  $\epsilon$  centered at  $\mathbf{x}$ .
- $A_{\epsilon,h}(\mathbf{x}) \subset B_\epsilon(\mathbf{x})$  denotes a generic discrete (but not necessarily lattice aligned) neighborhood of radius  $\epsilon$  surrounding the pixel  $\mathbf{x}$  and used for inpainting.
- $\text{Supp}(A_{\epsilon,h}(\mathbf{x})) \subset \mathbb{Z}_h^2$  denotes the set of real pixels needed to define  $A_{\epsilon,h}(\mathbf{x})$  based on bilinear interpolation.
- $B_{\epsilon,h}(\mathbf{x}) = \{\mathbf{y} \in \Omega_h : \|\mathbf{x} - \mathbf{y}\| \leq \epsilon\}$ , the discrete ball of radius  $\epsilon$  centered at  $\mathbf{x}$  and the choice of  $A_{\epsilon,h}(\mathbf{x})$  used by coherence transport.
- $r = \epsilon/h$  the radius of  $B_{\epsilon,h}(\mathbf{x})$  measured in pixels.
- $\tilde{B}_{\epsilon,h}(\mathbf{x}) = R(B_{\epsilon,h}(\mathbf{x}))$ , where  $R$  is the rotation matrix taking  $(0, 1)$  to  $\mathbf{g}(\mathbf{x})$ , the choice of  $A_{\epsilon,h}(\mathbf{x})$  used by Guidefill.
- $\mathcal{N}(\mathbf{x}) = \{\mathbf{x} + \mathbf{y} : \mathbf{y} \in \{-h, 0, h\} \times \{-h, 0, h\}\}$  is the 9-point neighborhood of  $\mathbf{x}$ .
- Given  $A_h \subset \mathbb{Z}_h^2$ , we define the dilation of  $A_h$  by

$$D_h(A_h) = \cup_{\mathbf{x} \in A_h} \mathcal{N}(\mathbf{x}).$$

If  $h = 1$  we write  $D$  instead of  $D_1$ .

- Given  $A_h \subset \mathbb{Z}_h^2$ , we define the discrete (inner) boundary of  $A_h$  by

$$\partial A_h := \{\mathbf{x} \in A_h : \mathcal{N}(\mathbf{x}) \cap \mathbb{Z}_h^2 \setminus A_h \neq \emptyset\}.$$

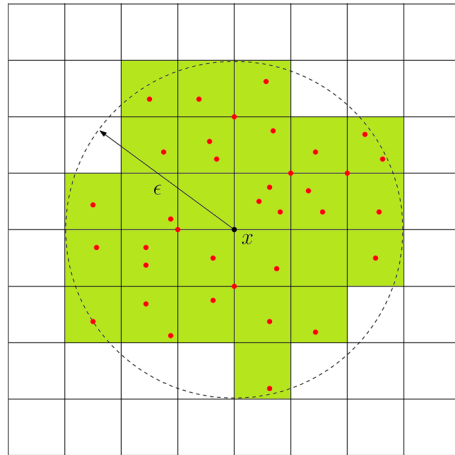
For convenience we typically drop the word “inner” and refer to  $\partial A_h$  as just the boundary of  $A_h$ .

- $O$  denotes the zero matrix.
- Given  $c \in \mathbb{R}$ , we define  $\{y \leq c\} := \{(x, y) \in \mathbb{R}^2 : y \leq c\}$ .
- Given  $L > 0$ , and  $0 \leq x, y < L$  we define the *circular distance* between  $x$  and  $y$  by

$$d_L^{\text{circle}}(x, y) = \min(x - y \bmod L, y - x \bmod L).$$

- Given  $\sigma, \mu \in \mathbb{R}$  we denote by  $G_{\sigma,\mu}$  and  $g_{\sigma,\mu}$  the Gaussians

$$G_{\sigma,\mu}(x) := \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma} \quad g_{\sigma,\mu}(x) := \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sum_{n \in \mathbb{Z}} e^{-\frac{(n-\mu)^2}{2\sigma^2}}}$$



**Fig. 3** Illustration of a generic set  $A_{\epsilon,h}(\mathbf{x})$  containing ghost pixels: in this illustration the overlaid grid is the lattice  $\mathbb{Z}_h^2$  with pixel centers at its vertices. The elements of a generic set  $A_{\epsilon,h}(\mathbf{x})$  are represented as red dots—they do not need to occupy pixel centers, but they must all lie within distance  $\epsilon$  of  $\mathbf{x}$ . Ghost pixels are defined based on bilinear interpolation of their real pixel neighbors. Here we have highlighted in green the squares whose vertices are the real pixels needed to define  $A_{\epsilon,h}(\mathbf{x})$ . We call this set of real pixels  $\text{Supp}(A_{\epsilon,h}(\mathbf{x}))$ . Note that while  $A_{\epsilon,h}(\mathbf{x}) \subset B_\epsilon(\mathbf{x})$ , this inclusion is not in general true of  $\text{Supp}(A_{\epsilon,h}(\mathbf{x}))$

Similarly, given  $L > 0$  and  $N \in \mathbb{N}$  we denote by  $G_{\sigma,\mu,L}^*$  and  $g_{\sigma,\mu,N}^*$  the periodic Gaussians

$$G_{\sigma,\mu,L}^*(x) := \frac{e^{-\frac{d_L^{\text{circle}}(x,\mu)^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma}$$

$$g_{\sigma,\mu,N}^*(x) := \frac{e^{-\frac{d_N^{\text{circle}}(x,\mu)^2}{2\sigma^2}}}{\sum_{n=0}^{N-1} e^{-\frac{d_N^{\text{circle}}(n,\mu)^2}{2\sigma^2}}}$$

Note that  $G_{\sigma,\mu}$  and  $g_{\sigma,\mu}$  are normalized with respect to integration over  $\mathbb{R}$  and summation over  $\mathbb{Z}$  respectively, while  $\frac{1}{\int_0^L G_{\sigma,\mu,L}^*(x)dx} \cdot G_{\sigma,\mu,L}^*$  and  $g_{\sigma,\mu,N}^*$  are normalized with respect to integration over  $[0, L)$  and summation over  $\{0, 1, \dots, N-1\}$  respectively.

## 2 Shell-Based Geometric Inpainting

In this paper we are interested in a simple class of shell-based geometric inpainting methods and their proposed semi-implicit extension. These methods fill the inpainting domain in successive shells from the boundary inwards, as illustrated in Fig. 1. In the direct form of the method, the color of a given pixel  $\mathbf{x}$  due to be filled is computed as a weighted average of its already filled neighbors within a discrete neighborhood

**Algorithm 1** Shell Based Geometric Inpainting

---

$u_h$  = damaged image, initialized to 0 on inpainting domain.  
 $\Omega = [a, b] \times [c, d]$  = continuous image domain.  
 $\Omega_h = \Omega \cap \mathbb{Z}_h^2$  = discrete image domain.  
 $D_h^{(0)}$  = initial inpainting domain.  
 $\partial D_h^{(0)}$  = initial inpainting domain boundary.  
 semiImplicit = false, unless we use the semi implicit extension (Section 5).  
**for**  $k = 0, \dots$  **do**  
   **if**  $D_h^{(k)} = \emptyset$  **then**  
     break  
   **end if**  
    $\partial_{\text{ready}} D_h^{(k)} = \{\mathbf{x} \in \partial D_h^{(k)} : \text{ready}(\mathbf{x})\}$   
    $u_h = \text{FILLBOUNDARY}(u_h, D_h^{(k)}, \partial_{\text{ready}} D_h^{(k)})$   
    $D_h^{(k+1)} = D_h^{(k)} \setminus \partial_{\text{ready}} D_h^{(k)}$   
   **if** semiImplicit **then**  
      $u_h^{(0)} = u_h$   
     **for**  $n = 1, 2, \dots$  (until convergence) **do**  
        $u_h^{(n)} = \text{FILLBOUNDARY}(u_h^{(n-1)}, D_h^{(k+1)}, \partial_{\text{ready}} D_h^{(k)})$   
     **end for**  
   **end if**  
    $\partial D_h^{(k+1)} = \{\mathbf{x} \in \partial D_h^{(k+1)} : \mathcal{N}(\mathbf{x}) \cap (\Omega_h \setminus D_h^{(k+1)}) \neq \emptyset\}$   
**end for**  
  
**function**  $u_h = \text{FILLBOUNDARY}(u_h, D_h, \partial D_h)$   
**for**  $\mathbf{x} \in \partial D_h$  **do**  
   compute  $A_{\epsilon, h}(\mathbf{x})$  = neighborhood of  $\mathbf{x}$ .  
   compute non-negative weights  $w_{\epsilon}(\mathbf{x}, \mathbf{y}) \geq 0$  for  $A_{\epsilon, h}(\mathbf{x})$ .  
   **if** ready( $\mathbf{x}$ ) **then**

$$u_h(\mathbf{x}) = \frac{\sum_{\mathbf{y} \in (A_{\epsilon, h}(\mathbf{x}) \setminus \{\mathbf{x}\}) \cap (\Omega \setminus D)} w_{\epsilon}(\mathbf{x}, \mathbf{y}) u_h(\mathbf{y})}{\sum_{\mathbf{y} \in (A_{\epsilon, h}(\mathbf{x}) \setminus \{\mathbf{x}\}) \cap (\Omega \setminus D)} w_{\epsilon}(\mathbf{x}, \mathbf{y})} \quad (1)$$
  
   **end if**  
**end for**  
**end function**

---

See (8) for a definition of the ready function for Guidefill. Coherence transport and Guidefill use the neighborhoods  $A_{\epsilon, h}(\mathbf{x}) = B_{\epsilon, h}(\mathbf{x})$ ,  $A_{\epsilon, h}(\mathbf{x}) = \tilde{B}_{\epsilon, h}(\mathbf{x})$  respectively - see Figure 10. They also both use the same weights (6). Blue text is only relevant for the semi-implicit extension we introduce in Section 5.

---

$A_{\epsilon, h}(\mathbf{x}) \subset B_{\epsilon}(\mathbf{x})$ . In the semi-implicit extension, this sum also includes unknown pixels within the current shell, resulting in a linear system (Fig. 2). We will cover the resulting linear system in detail in Sect. 5, where we also propose an iterative method for its solution. The direct method as well as this proposed iterative solution to the semi-implicit extension are illustrated in Algorithm 1 with pseudo code (the blue code indicates the parts relevant to the semi-implicit extension). The neighborhood  $A_{\epsilon, h}(\mathbf{x})$  need not be axis aligned and may contain “ghost pixels” lying between pixel centers—see Fig. 3 for an illustration. Ghost pixels were introduced in [27] (where they were shown to be beneficial for reducing “kinking artifacts”—see Fig. 6), and the color of a given ghost pixel is defined as the bilinear interpolation of its four real pixel neighbors,



but is undefined if one or more of them has not yet been assigned a color. We denote by  $\text{Supp}(A_{\epsilon,h}(\mathbf{x})) \subset \mathbb{Z}_h^2$  the set of real pixels needed to define  $A_{\epsilon,h}(\mathbf{x})$  in this way. Here  $h$  and  $\epsilon$  denote respectively the width of one pixel and the radius of the bounding disk  $B_\epsilon(\mathbf{x}) \supset A_{\epsilon,h}(\mathbf{x})$ . The averaging weights  $w_\epsilon$  are non-negative and are allowed to depend on  $\mathbf{x}$ , but must scale proportionally with the size of the neighborhood  $A_{\epsilon,h}(\mathbf{x})$ , like so:

$$w_\epsilon(\mathbf{x}, \mathbf{y}) = \hat{w}\left(\mathbf{x}, \frac{\mathbf{y} - \mathbf{x}}{\epsilon}\right) \quad (2)$$

for some function  $\hat{w}(\cdot, \cdot) : \Omega \times B_1(\mathbf{0}) \rightarrow [0, \infty]$ . Note that we will sometimes write  $w_r$  or  $w_1$  in place of  $w_\epsilon$ —in this case we mean (2) with  $\epsilon$  replaced by  $r$  or  $1$  in the denominator on the right hand side. As the algorithm proceeds, the inpainting domain shrinks, generating a sequence of inpainting domains

$$D_h = D_h^{(0)} \supset D_h^{(1)} \supset \dots \supset D_h^{(K)} = \emptyset.$$

We will assume the non-degeneracy condition

$$\sum_{\mathbf{y} \in A_{\epsilon,h}(\mathbf{x}) \cap (\Omega \setminus D^{(k)})} w_\epsilon(\mathbf{x}, \mathbf{y}) > 0 \quad (3)$$

holds at all times, this ensures that the average (1) in Algorithm 1 is always well defined. One trivial way of ensuring this is by having strictly positive weights  $\hat{w}$ , which all the methods considered do (see Sect. 3). At iteration  $k$ , only pixels belonging to the current boundary  $\partial D_h^{(k)}$  are filled, but moreover we fill only a subset  $\partial_{\text{ready}} D_h^{(k)} \subseteq \partial D_h^{(k)}$  of pixels deemed to be “ready” (In Sect. 3 we will review the main methods in the literature and give their “ready” functions). The main inpainting methods in the literature of the general form given by Algorithm 1 are (in chronological order) Telea’s Algorithm [38], coherence transport [7], coherence transport with adapted distance functions [31], and our previous work Guidefill [27]. As we will see, these methods essentially differ only in the choice of weights (2), the choice of fill order as dictated by the “ready” function, and the choice of neighborhood  $A_{\epsilon,h}(\mathbf{x})$ . We will review these methods in Sect. 3.

**Remark 2** It is worth mentioning that this class of algorithms is nearly exactly the same as the “generic single-pass algorithm” first systematically studied by Bornemann and März in [7]. The two main differences are

1. They assume  $A_{\epsilon,h}(\mathbf{x}) = B_{\epsilon,h}(\mathbf{x})$ , while we allow for greater flexibility.
2. They consider only the direct form of Algorithm 1, not the semi-implicit extension.

Beyond this, they also phrase things in terms of a pre-determined fill order, rather than a “ready” function, but the former may easily be seen, mathematically at least, to be a special case of the latter (Sect. 3).

## 2.1 Advantages of Algorithm 1

The main appeal of Algorithm 1 is its simplicity and parallelizability, which enable it to run very fast. A second advantage, first noted by Bornemann and März [7], is the stability property

$$\min_{\mathbf{y} \in B} u_0(\mathbf{y}) \leq u_h(\mathbf{x}) \leq \max_{\mathbf{y} \in B} u_0(\mathbf{y}) \quad \text{for all } \mathbf{x} \in D_h, \quad (4)$$

(which holds channelwise) where  $u_0 : \Omega_h \setminus D_h \rightarrow \mathbb{R}^d$  is the given image and  $B$  is the band of width  $\epsilon$  pixels surrounding  $\partial D_h$ . This property holds because we have chosen non-negative weights summing to one.

**Remark 3** Although we have presented Telea’s algorithm [38] as an example of Algorithm 1, this is not strictly true as its update formula (5) (see Sect. 3) contains a gradient term that, after it has been approximated with finite differences, effectively violates the rule of non-negative weights summing to one. This means that Telea’s algorithm does not satisfy the stability property (4). See Fig. 7.

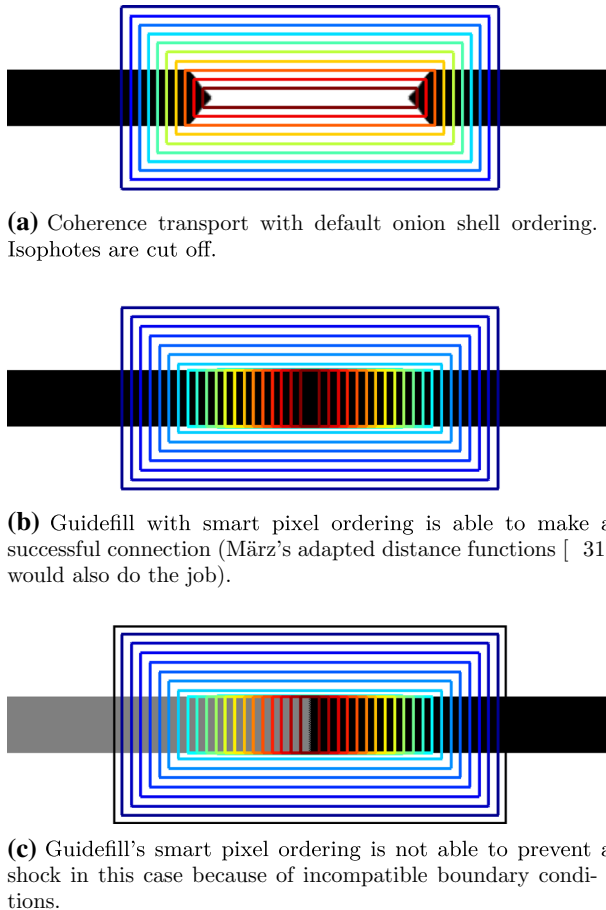
## 2.2 Disadvantages and Artifacts

The main disadvantage of Algorithm 1 is obviously loss of texture, and in cases where texture is important, these methods should not be used. However, beyond loss of texture, inpainting methods of the general form given in Algorithm 1 can also introduce a host of other artifacts, which we list below.

- **“kinking” of isophotes** where extrapolated isophotes change direction at the boundary of the inpainting domain—see Figs. 6 and 8.
- **“blurring” of isophotes** where edges that are sharp in the undamaged region may blur when extended into the inpainting domain—see Figs. 6 and 8.
- **“cutting off” of isophotes** where isophotes extrapolated into the inpainting domain end abruptly—see Fig. 4.
- **formation of shocks** where sharp discontinuities may form in the inpainting domain—see Fig. 4.
- **bright or dark spots** that are only a problem if the stability condition (4) is violated, as it is for Telea’s algorithm. See Figs. 7 and 8.

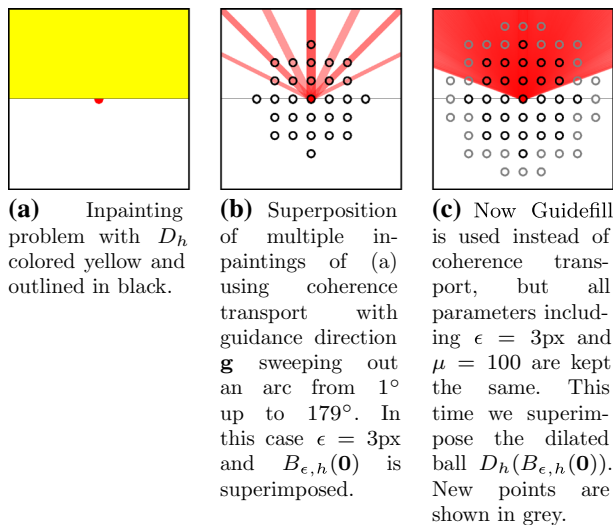
## 2.3 Related Work (Artifact Reduction)

Broadly speaking, there has been incremental progress as follows: Telea’s algorithm [38], the earliest variant to appear in the literature, suffers from strong artifacts of every type. In particular, the weights make no attempt to take into account the orientation of undamaged isophotes in  $\Omega_h \setminus D_h$ , and the result shows strong kinking artifacts (see Fig. 6). Bornemann and März identified and sought to address this problem with coherence transport [7], which proposed carefully chosen weights that are proven (in a high resolution and vanishing viscosity limit) to extend isophotes in any desired guidance direction  $\mathbf{g}$  not parallel to the inpainting domain boundary. This was



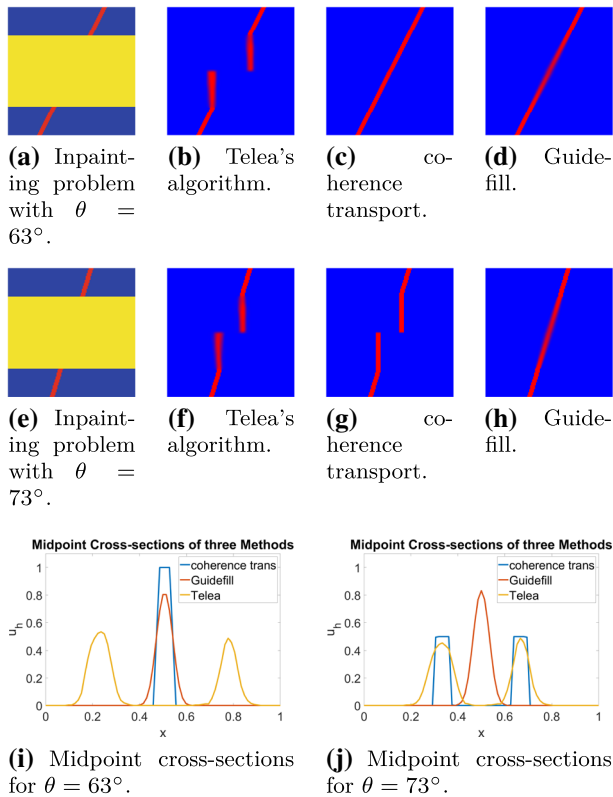
**Fig. 4** Cut off isophotes and shocks: because Algorithm 1 fills the inpainting domain from many directions at once, “cut off isophotes” or shocks can sometimes be formed. In **a** this is due to the (superimposed) fill order, which is the default onion shell ordering and a bad choice in this case. In **b** we have a chosen a new fill order better adapted to the image and the problem is solved in this case. However, the shock in **c** is due to incompatible boundary conditions and it is unlikely any special fill order could solve the problem

combined with a method aimed at robustly measuring the orientation of isophotes at the boundary, so that a suitable  $\mathbf{g}$  allowing for a seamless transition could be found. The problem of “kinking” ostensibly resolved, in a follow up work März proposed coherence transport with adapted distance functions [31] designed to minimize the problem of “cut off” isophotes and shocks. This was accomplished by recognizing that artifacts such as the incomplete line in Fig. 4a are often the byproduct of a sub-optimal fill order such as the one superimposed (in this case the default onion shell ordering). The situation can often be corrected as in Fig. 4b, by using an ordering better adapted to the image such as the one illustrated there. Rather than filling pixels in an order proportional to their distance from the boundary, i.e. having the ready function in Algorithm 1 always return “true”, März proposed a number of ways of generating improved orderings based on non-Euclidean distance from boundary maps.



**Fig. 5** Special directions: for a given guidance direction  $\mathbf{g} = (\cos \theta, \sin \theta)$ , coherence transport [7,31] can successfully extrapolate isophotes parallel to  $\mathbf{g}$  only if  $\mathbf{g} = \lambda \mathbf{v}$ , for some  $\mathbf{v} \in B_{\epsilon,h}(\mathbf{0})$ . This is illustrated in **b**, where we have solved the inpainting problem posed in **a** multiple times using coherence transport with  $\epsilon = 3\text{px}$  with a sequence of guidance directions  $\mathbf{g}_k = (\cos \theta_k, \sin \theta_k)$  ( $\theta_k \in [k^\circ]_{k=1}^{179}$ ) and superimposed the results together with  $B_{\epsilon,h}(\mathbf{0})$  (the parameter  $\mu$  in (6) is  $\mu = 100$ ). Instead of a smoothly varying line sweeping through the upper half plane and filling it with red, we see a superposition of finitely many lines, each passing through some  $\mathbf{v} \in B_{\epsilon,h}(\mathbf{0})$ . When we repeat the experiment in **c** using Guidefill [27], we see that it is not free of problems either. In this case Guidefill can extrapolate along  $\mathbf{g} = (\cos \theta, \sin \theta)$  so long as  $0 < \theta_c \leq \theta \leq \pi - \theta_c < \pi$ , where  $\theta_c$  is a critical angle, and we get a red cone bounded on either side by  $\theta_c$ . Here we have superimposed the dilated ball  $D_h(B_{\epsilon,h}(\mathbf{0}))$ , and it is evident that  $\theta_c$  is in some way related to this dilation—this will be explained in Sect. 7.1.1

At the same time, recognizing that the presence of shocks was related to the “stopping set” [31] of the distance map, März was able to exert some measure of control over those as well, if not prevent them entirely. Guidefill [27] brought the focus back to the reduction of kinking artifacts, by noting that coherence transport is actually only able to propagate along a given guidance vector  $\mathbf{g}$  if it points in one of a finite set of special directions—see Fig. 5b. Whereas previous improvements to Algorithm 1 had focused first on improving the choice of weights, then the fill order (equivalently the choice of ready function), Guidefill proposed for the first time to change the averaging neighborhood  $A_{\epsilon,h}(\mathbf{x})$ , which until then had always been the discrete ball  $B_{\epsilon,h}(\mathbf{x})$  (Fig. 10a). Specifically, it proposed to replace  $A_{\epsilon,h}(\mathbf{x}) = B_{\epsilon,h}(\mathbf{x})$  with  $A_{\epsilon,h}(\mathbf{x}) = \tilde{B}_{\epsilon,h}(\mathbf{x})$ , where  $\tilde{B}_{\epsilon,h}(\mathbf{x})$  is the *rotated* discrete ball shown in Fig. 10b, aligned with the guidance direction  $\mathbf{g}$ . Since  $A_{\epsilon,h}(\mathbf{x})$  is in this case no longer axis aligned, it contains what the authors called “ghost pixels” lying between pixel centers, which they defined based on bilinear interpolation. This small change enabled Guidefill to propagate along most guidance directions, but it too has problems when the angle between  $\mathbf{g}$  and the boundary to the inpainting domain is too shallow—see Fig. 5c. However, Guidefill pays a price for its reduction in kinking artifacts in the form of an increase in blur artifacts. See Fig. 6, where coherence transport produces a sharp extension of image isophotes, albeit possibly in the wrong direction, whereas Guidefill extrapolates in the right direction, but



**Fig. 6** A tale of two inpainting problems: in **a–d** a line making an angle of  $\theta = 63^\circ$  with the horizontal is inpainted using each of Telea’s algorithm [38], coherence transport, [7,31], and Guidefill [27] (the inpainting domain is shown in yellow). In this case the radius of  $A_{\epsilon,h}(\mathbf{x})$  is  $\epsilon = 3\text{px}$ , and since  $63^\circ \approx \arctan(2) \approx 63.44^\circ$  is close to one of the “special directions” in which coherence transport can extend isophotes successfully for this value of  $\epsilon$  (see Fig. 5), both coherence transport and Guidefill make a successful connection. In **e–h** we change the angle of the line slightly to  $\theta = 73^\circ$ . This isn’t one of coherence transport’s admissible directions for  $\epsilon = 3\text{px}$ , so it fails to make the connection, while Guidefill continues to have no problems, at the expense of some blurring. Telea’s algorithm, on the other hand, propagates in the direction of the normal to the inpainting domain boundary regardless of the undamaged image content, and thus fails to make the connection in both cases while also introducing significant blur. In **i, j**, we examine horizontal cross sections (of the red channel) of all three methods at the midpoint of the inpainting domain. Here, a disadvantage of Guidefill in terms of blur becomes more apparent—coherence transport by contrast produces a much sharper result. The reasons for this are explored in Sect. 7.3

the extrapolation suffers from blur. Guidefill also proposed its own “smart order” computed on the fly as an alternative to März’s adapted distance functions, but this does not have any serious advantage in terms of the quality of the results. Either approach will do for preventing “cut off” isophotes.

## 2.4 Related Theoretical Work

The direct form of Algorithm 1 has been studied from a theoretical point of view by proposing two distinct continuum limits. The first of these is the high-resolution

vanishing viscosity limit proposed by Bornemann and März, in which  $h \rightarrow 0$  and then  $\epsilon \rightarrow 0$  [7]. The second is a fixed-ratio continuum limit proposed in our previous work [27] in which  $(h, \epsilon) \rightarrow (0, 0)$  along the line  $\epsilon = rh$  (with  $r \in \mathbb{N}$  fixed). The non-negative integer  $r$  is simply the radius of  $A_{\epsilon,h}(\mathbf{x})$  measured in pixels. Although both are perfectly valid mathematically, numerical experiments indicate that the high resolution viscosity limit gives a good approximation of the behaviour of Algorithm 1 in practice only when  $r \gg 1$ , whereas our fixed-ratio limit gives a good approximation even when  $r$  is a small integer, as it typically is in practice (see Remark 4.3 in our previous work [27] for an explanation of why this is). There has also been significant work in studying the well-posedness of the high resolution and vanishing viscosity limit of Algorithm 1, both in [7] and especially in [32]. See Fig. 9 for an illustration of these two separate limits.

**Motivation for ghost pixels** In Sect. 4 we will prove that any weighted sum over a set  $A_{\epsilon,h}(\mathbf{x})$  of ghost pixels is equivalent to a sum over the real pixels in  $\text{Supp}(A_{\epsilon,h}(\mathbf{x}))$  with equivalent weights. While this makes ghost pixels in some sense redundant, they are useful concept. Specifically, in Theorem 6 we will prove that the fixed-ratio continuum limit described above and illustrated in Fig. 9 is a transport equation with transport direction that is a function of the weights  $w_\epsilon$  and the position vectors of the elements of  $A_{\epsilon,h}(\mathbf{x})$ . Avoiding “kinking” artifacts amounts to working backwards from a desired transport direction  $\mathbf{g}$  to the weights  $w_\epsilon$  and neighborhood  $A_{\epsilon,h}(\mathbf{x})$  that yield  $\mathbf{g}$ . This is easier to do when the elements of  $A_{\epsilon,h}(\mathbf{x})$  can move continuously in  $\mathbb{R}^2$ , rather than being constrained to the lattice  $\mathbb{Z}_h^2$ .

## 2.5 Contributions

Our contributions are both theoretical and practical, aimed at a deeper understanding of the mathematical properties of both the direct and semi-implicit versions of Algorithm 1, and through that understanding, a better grasp of the underlying causes of the artifacts described above and what, if anything, can be done about them. Our main targets are “kinking” and “blurring” artifacts, the others having already been thoroughly analyzed in [7,32] and well understood. Broadly speaking, we have three main contributions:

- We propose a semi-implicit variant of Algorithm 1, propose an efficient method for solving the linear system that arises in it, and derive analytical rates of convergence.
- We propose two novel limits of Algorithm 1: a fixed-ratio continuum limit where the pixel width  $h \rightarrow 0$  with  $r = \epsilon/h$  fixed, and an asymptotic limit where  $r = \epsilon/h$  is again fixed but rather than taking  $h$  to zero, we explore the asymptotic dependence of the solution on  $h$  when it is very small.
- We use the above two limits to explain the origins of kinking and blur artifacts in Algorithm 1, and how to rectify them. Among other things, we prove that *any* variant of the direct form of Algorithm 1 must exhibit kinking artifacts, whereas this is not the case for our proposed semi-implicit extension (Sect. 7.1.1).

While our present work focuses on inpainting, our results—in particular the asymptotic limit—are abstract results that can be applied more generally. An example of a situation where this comes up is the use of damped Jacobi iteration for solving a linear system

$$A\mathbf{x} = \mathbf{b}$$

when  $A$  is an M-matrix (that is, has a positive diagonal and negative or zero off-diagonal elements). Section 8 demonstrates this by applying our asymptotic limit to predict the evolution of the error in damped Jacobi applied to a 1-d convection-diffusion problem with Dirichlet boundary conditions.

### 3 Review of Main Methods

Here we briefly review the main inpainting methods of the general form sketched in Algorithm 1.

**Telea’s algorithm** The earliest algorithm (to our knowledge) appearing in the literature and of the form sketched in Algorithm 1, Telea’s algorithm [38] is also the only such algorithm to use a different formula for  $u_h(\mathbf{x})$  than the expression (1) appearing in Algorithm 1 (see Remark 3). Instead of computing  $u_h(\mathbf{x})$  a weighted average of  $u_h(\mathbf{y})$  evaluated at nearby already filled pixels  $\mathbf{y}$ , it takes a weighted average of the *predictions* that each of these pixels makes, based on linear extrapolation, for  $u_h(\mathbf{x})$ . That is,

$$u_h(\mathbf{x}) = \frac{\sum_{\mathbf{y} \in B_{\epsilon,h}(\mathbf{x}) \cap (\Omega_h \setminus D_h^{(k)})} w_{\epsilon}(\mathbf{x}, \mathbf{y}) u_{\text{predicted}}(\mathbf{x})}{\sum_{\mathbf{y} \in B_{\epsilon,h}(\mathbf{x}) \cap (\Omega_h \setminus D_h^{(k)})} w_{\epsilon}(\mathbf{x}, \mathbf{y})}, \quad (5)$$

where

$$u_{\text{predicted}}(\mathbf{x}) := u_h(\mathbf{y}) + \nabla_h u_h(\mathbf{y}) \cdot (\mathbf{x} - \mathbf{y})$$

and  $\nabla_h u_h(\mathbf{y})$  denotes the centered difference approximation to the gradient of  $u_h$  at  $\mathbf{y}$ , that is

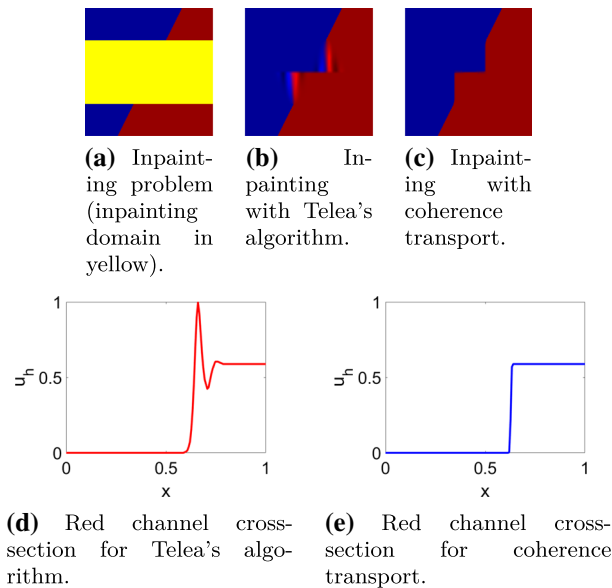
$$\begin{aligned} \nabla_h u_h(\mathbf{y}) := & \frac{1}{2} (u_h(\mathbf{y} + \mathbf{e}_1) \\ & - u_h(\mathbf{y} - \mathbf{e}_1), u_h(\mathbf{y} + \mathbf{e}_2) - u_h(\mathbf{y} - \mathbf{e}_2)). \end{aligned}$$

As we have already noted in Remark 3, this approach has a disadvantage in that it results in the loss of the stability property (4). Moreover, the predictions based on linear extrapolation can become highly inaccurate when  $\mathbf{y}$  is on an edge in  $\Omega_h \setminus D_h^{(k)}$ , leading to significant over or undershooting, visible as bright or dark spots as in Figs. 7 and 8. Perhaps in recognition of this, the gradient term was dropped from (5) in all subsequent algorithms. The weights in this case are

$$w_{\epsilon}(\mathbf{x}, \mathbf{y}) = \text{dir}(\mathbf{x}, \mathbf{y}) \cdot \text{dst}(\mathbf{x}, \mathbf{y}) \cdot \text{lev}(\mathbf{x}, \mathbf{y}),$$

where

$$\text{dir}(\mathbf{x}, \mathbf{y}) := \frac{\mathbf{x} - \mathbf{y}}{\|\mathbf{x} - \mathbf{y}\|} \cdot \mathbf{N}(\mathbf{x}),$$



**Fig. 7** Bright spots in Telea's algorithm: in this example we consider the inpainting problem shown in **a** consisting of a line separating a region of dark blue from a region of dark red. We inpaint both using Telea's algorithm (**b**) and coherence transport (**c**). Coherence transport obeys the stability property (4) and hence the brightness of the inpainted solution remains bounded above by the brightness on the exterior of the inpainting domain. This is not true of Telea's algorithm, which exhibits bright spots outside the original color range. These were not visible in Fig. 6, because the brightness of each color channel was already saturated, and Telea's algorithm uses clamping to prevent the solution from going outside the admissible color range. This is further illustrated in **d**, **e**, where we plot horizontal cross sections of the red channel of each inpainted solution

$$\text{dst}(\mathbf{x}, \mathbf{y}) := \frac{d_0^2}{\|\mathbf{x} - \mathbf{y}\|^2},$$

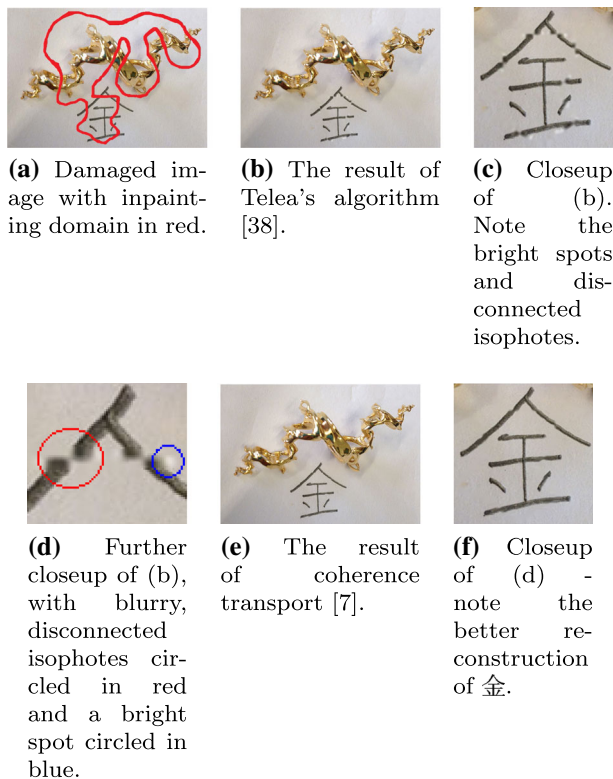
$$\text{lev}(\mathbf{x}, \mathbf{y}) := \frac{T_0}{1 + |T(\mathbf{y}) - T(\mathbf{x})|},$$

and  $T(\mathbf{x})$  denotes the Euclidean distance from  $\mathbf{x}$  to the (original) boundary of the inpainting domain, and  $N(\mathbf{x}) = \nabla_h T(\mathbf{x})$  (estimated based on central differences).  $T$  is precomputed using the fast marching method. Telea's algorithm uses the default onion shell ordering, that is “ready( $\mathbf{x}$ )  $\equiv$  true” (Fig. 9).

**Coherence transport** Coherence transport [7] improves upon Telea's algorithm by adapting the weights in order to encourage extrapolation of isophotes in the direction of their tangent. This is done by calculating a “local coherence direction”  $\mathbf{g}(\mathbf{x})$  in terms of a modified structure tensor. Coherence transport calculates the color of a given pixel to be filled using the formula (1) in Algorithm 1 with weights

$$w_\epsilon(\mathbf{x}, \mathbf{y}) = \frac{1}{\|\mathbf{y} - \mathbf{x}\|} \exp\left(-\frac{\mu^2}{2\epsilon^2} (\mathbf{g}^\perp(\mathbf{x}) \cdot (\mathbf{y} - \mathbf{x}))^2\right), \quad (6)$$





**Fig. 8** Blurring, kinking, and bright spots with Telea's algorithm: even for problems such as **a** where the inpainting domain is very thin, Telea's algorithm **b–d** still creates strong blurring artifacts and fails to connect isophotes effectively. Also, due to the presence of the gradient term in (5), Telea's algorithm violates the stability condition (4) and as a result can “overshoot” when filling pixels close to edges in the filled area, where the (numerical) gradient changes rapidly. This leads to the bright spots near the reconstructed in **c, d**. In this case coherence transport **e, f** is a much better choice

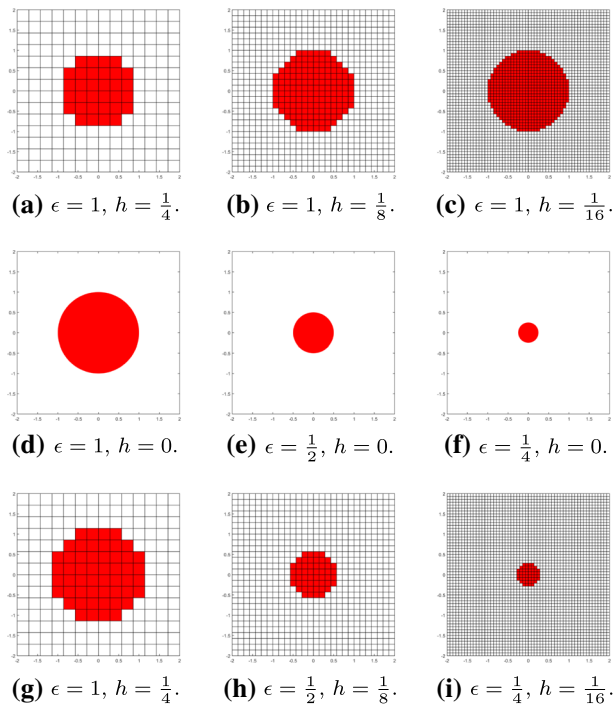
and with  $A_{\epsilon,h}(\mathbf{x}) = B_{\epsilon,h}(\mathbf{x})$ —see Fig. 10a and c. Like Telea's algorithm, coherence transport uses the default onion shell ordering, that is “ready( $\mathbf{x}$ )  $\equiv$  true”.

**Coherence transport with adapted distance functions** In a subsequent work [31], März made improvements to coherence transport by replacing the default onion shell ordering with one based on a variety of non-Euclidean distance functions. One such distance function defines an “active boundary”  $\Gamma_h \subseteq \partial D_h$  defined by

$$\Gamma_h := \{\partial D_h : \langle \mathbf{g}(\mathbf{x}), \mathbf{N}(\mathbf{x}) \rangle^2 > \gamma\}$$

where  $\gamma > 0$  is a small constant. The non-Euclidean distance to boundary  $T_h^*$  is then computed as the Euclidean distance to the active boundary. The algorithm is modified so that at any given iteration, only a subset of boundary pixels are filled—namely those minimizing  $T_h^*$ . That is

$$\text{ready}(\mathbf{x}) = \text{true} \Leftrightarrow \mathbf{x} \in \operatorname{argmin}_{\mathbf{y} \in \partial D_h} T_h^*(\mathbf{y}).$$

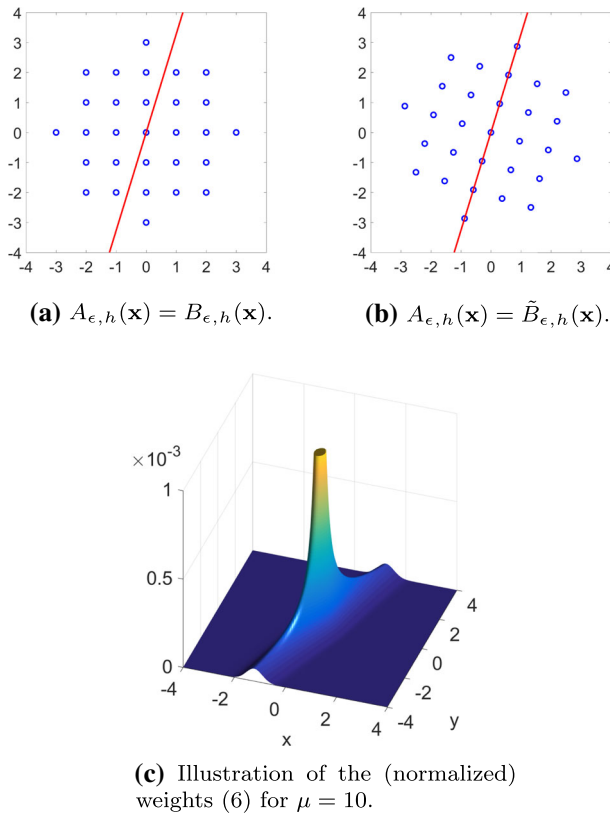


**Fig. 9** Two distinct continuum limits: Algorithm 1 has two distinct continuum limits, illustrated here for  $A_{\epsilon,h}(\mathbf{x}) = B_{\epsilon,h}(\mathbf{x})$ . The first, illustrated in **a–f**, is the high-resolution vanishing viscosity double limit proposed by Bornemann and März [7], in which  $h \rightarrow 0$  (**a**)–(**c**) and then  $\epsilon \rightarrow 0$  (**d**)–(**f**). The second is the fixed-ratio limit single limit  $(\epsilon, h) \rightarrow (0, 0)$  with  $r = \frac{\epsilon}{h}$  fixed proposed in our previous work [27], illustrated in **g–i** for  $r = 4$ . While they are both valid limits of Algorithm 1, they predict very different behaviour. In particular, while the high-resolution vanishing viscosity of Bornemann and März [7] is able to predict the “kinking” behaviour of Telea’s Algorithm [38], it fails to predict the kinking artifacts of their own method, coherence transport. Our fixed-ratio continuum limit, on the other hand, predicts both. See Theorem 6, as well as Sect. 7.1.2

This adaptation leads to improvements in the long range extrapolation of isophotes, as in Fig. 4.

**Guidefill** Guidefill [27] is a recent inpainting algorithm designed to address, among other things, the kinking issues in Figs. 5b and 6. While coherence transport is able to extrapolate along guidance direction  $\mathbf{g}(\mathbf{x})$  only if  $\mathbf{g}(\mathbf{x}) = \lambda(\mathbf{v} - \mathbf{x})$  for some  $\mathbf{v} \in B_{\epsilon,h}(\mathbf{x})$  (see Fig. 5b), Guidefill replaces the lattice aligned discrete ball  $B_{\epsilon,h}(\mathbf{x})$  with the *rotated discrete ball*  $\tilde{B}_{\epsilon,h}(\mathbf{x})$  aligned with the local transport direction  $\mathbf{g}(\mathbf{x})$ , so that  $\mathbf{g}(\mathbf{x}) = \lambda(\mathbf{v} - \mathbf{x})$  for some  $\mathbf{v} \in \tilde{B}_{\epsilon,h}(\mathbf{x})$  is *always* true. The rotated ball  $\tilde{B}_{\epsilon,h}(\mathbf{x})$  contains “ghost pixels” lying between pixel centers which are defined using bilinear interpolation. See Sect. 4 for a deeper discussion of ghost pixels, as well as Fig. 10a–b for an illustration of  $B_{\epsilon,h}(\mathbf{x})$  and  $\tilde{B}_{\epsilon,h}(\mathbf{x})$ .

Guidefill uses the same weights (6) as coherence transport (illustrated in Fig. 10c) and similarly to the latter’s extension [31], it has a way of automatically determining a good fill order. Unlike coherence transport which computes  $\mathbf{g}(\mathbf{x})$  concurrently with inpainting, Guidefill computes a guide field  $\mathbf{g}(\mathbf{x}) : D_h \rightarrow \mathbb{R}^2$  prior to inpainting. The



**Fig. 10** Neighborhoods and weights for coherence transport and Guidefill: here we illustrate the neighborhoods  $A_{\epsilon,h}(\mathbf{x})$  and weights (6) used by coherence transport and Guidefill. In each case  $\epsilon = 3\text{px}$  and  $\mathbf{g}(\mathbf{x}) = (\cos 73^\circ, \sin 73^\circ)$ . Coherence transport **a** uses the lattice-aligned discrete ball  $A_{\epsilon,h}(\mathbf{x}) = B_{\epsilon,h}(\mathbf{x})$ , while Guidefill **b** uses the rotated discrete ball  $A_{\epsilon,h}(\mathbf{x}) = \tilde{B}_{\epsilon,h}(\mathbf{x})$ . The ball  $\tilde{B}_{\epsilon,h}(\mathbf{x})$  is rotated so that it is aligned with the line  $L$  (shown in red) passing through  $\mathbf{x}$  parallel to  $\mathbf{g}(\mathbf{x})$ . In general  $\tilde{B}_{\epsilon,h}(\mathbf{x})$  contains “ghost pixels” lying between pixel centers, which are defined using bilinear interpolation of their “real” pixel neighbors. Both use the same weights (6) illustrated in **c**. The parameter  $\mu$  controls the extent to which the weights are biased in favor of points lying on or close to the line  $L$

guide field is computed based on splines which the user may adjust in order to influence the results. It is used to automatically compute a good fill order by computing for each  $\mathbf{x} \in \partial D_h$  a confidence  $C(\mathbf{x}) \in [0, 1]$  inspired by Criminisi et al. [16] and given by

$$C(\mathbf{x}) = \frac{\sum_{\mathbf{y} \in \tilde{B}_{\epsilon,h}(\mathbf{x}) \cap (\Omega \setminus D^{(k)})} w_\epsilon(\mathbf{x}, \mathbf{y})}{\sum_{\mathbf{y} \in \tilde{B}_{\epsilon,h}(\mathbf{x})} w_\epsilon(\mathbf{x}, \mathbf{y})}, \quad (7)$$

and then only filling those pixels for which  $C(\mathbf{x}) > c$ , where  $c \in (0, 1)$  is a small constant. That is

$$\text{ready}(\mathbf{x}) = 1(C(\mathbf{x}) > c) \quad (8)$$

Guidefill was designed for use as part of a 3D conversion pipeline, and as such makes use of a set  $B_h$  of “bystander pixels” which are neither inpainted nor may be used for inpainting. However, this is not relevant to our current investigation and we will assume  $B_h = \emptyset$  throughout. As shown in Fig. 5c—Guidefill is able to largely, but not completely, eliminate kinking artifacts. It was in the hope of overcoming this that we designed the semi-implicit version of Algorithm 1 discussed in Sect. 5.

**Remark 4** Note that we have deliberately excluded the point  $\mathbf{x}$  from the update formula (1) in Algorithm 1, even if the set  $A_{\epsilon,h}(\mathbf{x})$  contains  $\mathbf{x}$ . This is *not* done in any of the methods [7,27,31,38] we have just discussed, but it makes no difference to them or any other variant of the direct form of Algorithm 1, because the subroutine FillRow only involves sums taken over  $A_{\epsilon,h}(\mathbf{x}) \cap (\Omega \setminus D^{(k)})$ , which never contains  $\mathbf{x}$ . However, the semi-implicit extension of Algorithm 1 expresses  $u_h(\mathbf{x})$  as a sum of  $u_h(\mathbf{y})$  over a set of points that might include  $\mathbf{x}$ . This creates problems with weights such as (6) for which  $w_\epsilon(\mathbf{x}, \mathbf{x}) = \infty$ . See “Appendix A” for further details.

## 4 Ghost Pixels and Equivalent Weights

Because ghost pixels are defined using bilinear interpolation, any sum over a finite set of ghost pixels  $A(\mathbf{x})$  can be converted into a sum over an equivalent set of real pixels with equivalent weights,<sup>1</sup> that is

$$\sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) u_h(\mathbf{y}) = \sum_{\mathbf{y} \in \text{Supp}(A(\mathbf{x}))} \tilde{w}(\mathbf{x}, \mathbf{y}) u_h(\mathbf{y})$$

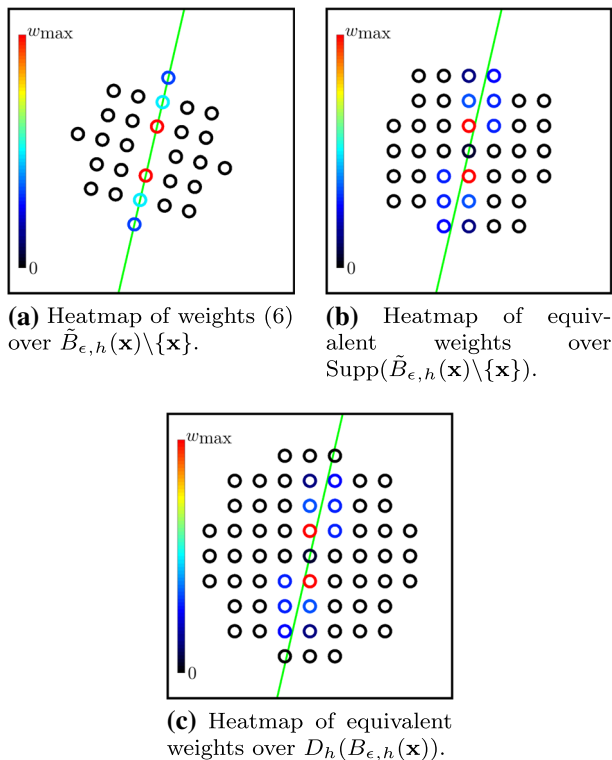
where  $\text{Supp}(A(\mathbf{x}))$  denotes the set of real pixels needed to define  $u_h(\mathbf{y})$  for each  $\mathbf{y} \in A(\mathbf{x})$  and  $\tilde{w}$  denotes a set of equivalent weights. This works because each  $u_h(\mathbf{y})$  is itself a weighted sum of the form

$$u_h(\mathbf{y}) = \sum_{\mathbf{z} \in \mathbb{Z}_h^2} \Lambda_{\mathbf{z},h}(\mathbf{y}) u_h(\mathbf{z}),$$

where  $\{\Lambda_{\mathbf{z},h}\}_{\mathbf{z} \in \mathbb{Z}_h^2}$  denote the basis functions of bilinear interpolation associated with the lattice  $\mathbb{Z}_h^2$ . This is illustrated in Fig. 11a, b, where we show a heat map of the weights (6) over the set  $\tilde{B}_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\}$  for  $\mu = 100$  and  $\epsilon = 3\text{px}$ , as well as a similar heat map of the modified weights over  $\text{Supp}(\tilde{B}_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\}) \subseteq D_h(B_{\epsilon,h}(\mathbf{x}))$ . Note that even though  $\tilde{B}_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\}$  does not contain the point  $\mathbf{x}$ , the support of this set does. This will be important in Sect. 5. Here we briefly list some properties of equivalent weights, including an explicit formula. Proofs are sketched, but details are deferred to “Appendix B”.

**Properties of equivalent weights** Properties 1–3 deal with a general finite set  $A(\mathbf{x})$  and general weights  $w(\mathbf{x}, \mathbf{y})$ , while properties 4–6 deal with the specific set  $A_{\epsilon,h}(\mathbf{x}) \subset B_\epsilon(\mathbf{x})$  and the specific weights  $w_\epsilon(\mathbf{x}, \mathbf{y})$  obeying (3).

<sup>1</sup> Note that here we mean a general family of finite sets  $A(\mathbf{x}) \in \mathbb{R}^2$  and general weights  $w(\mathbf{x}, \mathbf{y})$ . We do not mean the specific family of sets  $A_{\epsilon,h}(\mathbf{x})$  or the specific weights  $w_\epsilon(\mathbf{x}, \mathbf{y})$ , which have special properties.



**Fig. 11** Ghost pixels and equivalent weights: because ghost pixels are defined using bilinear interpolation, any weighted sum over a set of ghost pixels  $A_{\epsilon,h}(\mathbf{x})$  is equivalent to a sum with equivalent weights over real pixels in the set  $\text{Supp}(A_{\epsilon,h}(\mathbf{x}))$ , defined as the set of real pixels needed to define each ghost pixel  $\mathbf{y}$  in  $A_{\epsilon,h}(\mathbf{x})$ . We illustrate this in **a–c** using Guidefill with  $\epsilon = 3\text{px}$ ,  $\mathbf{g} = (\cos 77^\circ, \sin 77^\circ)$ , and  $\mu = 100$ . In **a**, the (normalized) weights (6) are visualized as a heat map over  $\tilde{B}_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\}$ . In **b** we show the equivalent weights over  $\text{Supp}(\tilde{B}_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\}) \subseteq D_h(B_{\epsilon,h}(\mathbf{x}))$  (this containment comes from (14) in Remark 6). Note that even though  $\tilde{B}_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\}$  does not contain the point  $\mathbf{x}$ , the support of this set does. In **c** we visualize the equivalent weights over the set  $D_h(B_{\epsilon,h}(\mathbf{x}))$ , which is strictly larger than  $\text{Supp}(\tilde{B}_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\})$ . For reference, we include the line parallel to  $\mathbf{g}$  in green

### 1. Explicit formula:

$$\tilde{w}(\mathbf{x}, \mathbf{z}) = \sum_{\mathbf{y} \in A(\mathbf{x})} \Lambda_{\mathbf{y},h}(\mathbf{z}) w(\mathbf{x}, \mathbf{y}). \quad (9)$$

### 2. Preservation of total mass:

$$\sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{y} \in \text{Supp}(A(\mathbf{x}))} \tilde{w}(\mathbf{x}, \mathbf{y}). \quad (10)$$

3. Preservation of center of mass (or first moment):

$$\sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) \mathbf{y} = \sum_{\mathbf{y} \in \text{Supp}(A(\mathbf{x}))} \tilde{w}(\mathbf{x}, \mathbf{y}) \mathbf{y}. \quad (11)$$

4. Inheritance of non-negativity:

$$\tilde{w}_\epsilon(\mathbf{x}, \mathbf{z}) \geq 0 \quad \text{for all } \mathbf{z} \in \text{Supp}(A_{\epsilon,h}(\mathbf{x})). \quad (12)$$

5. Inheritance of non-degeneracy condition (3):

$$\sum_{\mathbf{y} \in \text{Supp}(A_{\epsilon,h}(\mathbf{x}) \cap (\Omega \setminus D^{(k)}))} \tilde{w}_\epsilon(\mathbf{x}, \mathbf{y}) > 0. \quad (13)$$

6. Universal support: For any  $n \in \mathbb{Z}$ , we have

$$\begin{aligned} \text{Supp}(A_{\epsilon,h}(\mathbf{x}) \cap \{y \leq nh\}) &\subseteq D_h(B_{\epsilon,h}(\mathbf{x})) \cap \{y \leq nh\} \\ &\subseteq B_{\epsilon+2h,h}(\mathbf{x}) \cap \{y \leq nh\}. \end{aligned}$$

where  $\{y \leq nh\} := \{(x, y) \in \mathbb{R}^2 : y \leq nh\}$ , and where  $D_h$  is the dilation operator defined in our section on notation.

**Proof** Most of these properties are either obvious or are derived based on a simple exercise in changing the order of nested finite sums. Properties (10) and (11) are slightly more interesting—they follow from the fact that the bilinear interpolant of a polynomial of degree at most one is just the polynomial again. Note that an analogous formula for preservation of the second moment does *not* hold, because a quadratic function and its bilinear interpolant are not the same thing. The last identity is based on an explicit formula for the support of a point. Details are provided in “Appendix B”.  $\square$

**Remark 5** Although we have explicit formula (9) for the equivalent weights which will occasionally be useful, most of the time it is more fruitful to think about them in the following way: To compute  $\tilde{w}_\epsilon(\mathbf{x}, \mathbf{y})$  for some real pixel  $\mathbf{y}$ , loop over the ghost pixels  $\mathbf{z}$  such that  $\mathbf{y} \in \text{Supp}(\mathbf{z})$ . Then, each such  $\mathbf{z}$  redistributes to  $\tilde{w}_\epsilon(\mathbf{x}, \mathbf{y})$  a fraction of its weight  $w_\epsilon(\mathbf{x}, \mathbf{z})$  equal to the proportion of  $u_h(\mathbf{y})$  that went into  $u_h(\mathbf{z})$ .

**Remark 6** An obvious corollary of the universal support property (14) is that we also have the containment

$$\text{Supp}(A_{\epsilon,h}(\mathbf{x})) \subseteq D_h(B_{\epsilon,h}(\mathbf{x})) \subseteq B_{\epsilon+2h,h}(\mathbf{x}). \quad (14)$$

Figure 11 illustrates an example where this containment is strict, and in fact it is not hard to show that this holds in general. However, (14) is tight enough for our purposes in this paper.

**Remark 7** The main results of this paper, in particular Theorems 2 and 6, remain applicable if ghost pixels are defined using a different form of interpolation (rather than bilinear interpolation as above) so long as that interpolation scheme can also be expressed in terms of non-negative basis functions with compact support and also preserves polynomials of degree one. Moreover, the fixed-ratio continuum limit (Theorem 6) is *identical* for all such interpolation schemes. The asymptotic limit (Theorem 2) however, is not. One example of an alternative interpolation scheme obeying these properties is *Barycentric* triangle interpolation [10]. In the future, it would be interesting to explore alternative interpolation schemes for ghost pixels, however this is beyond the scope of the current work.

## 5 Semi-Implicit Extension of Algorithm 1

Here we present the semi-implicit form of Algorithm 1, to our knowledge not previously proposed in the literature, in which instead of computing  $u_h(\mathbf{x})$  for each  $\mathbf{x} \in \partial_{\text{ready}} D_h^{(k)}$  independently, we solve for them simultaneously by solving a linear system. We call our method semi-implicit in order to distinguish it from fully implicit methods in which the entire inpainting domain  $\{u_h(\mathbf{x}) : \mathbf{x} \in D_h\}$  is solved simultaneously, as is typically the case for most inpainting methods based on PDEs or variational principles, e.g. [6, 11, 13]. Specifically, we solve

$$\mathcal{L}\mathbf{u} = \mathbf{f} \quad \text{where } \mathbf{u} = \{u_h(\mathbf{x}) : \mathbf{x} \in \partial_{\text{ready}} D_h^{(k)}\}. \quad (15)$$

and  $\mathbf{f}$  is a vector of length  $|\partial_{\text{ready}} D_h^{(k)}|$ . The explicit entries of  $\mathcal{L}$  are written in terms of the equivalent weights  $\tilde{w}_\epsilon$  introduced in Sect. 4. Defining

$$S_{\epsilon,h}^{(k)}(\mathbf{x}) := \text{Supp}(A_{\epsilon,h}(\mathbf{x})) \cap \partial_{\text{ready}} D_h^{(k)},$$

it follows that  $\mathcal{L}$  couples each  $\mathbf{x} \in \partial_{\text{ready}} D_h^{(k)}$  to its immediate neighbors in  $S_{\epsilon,h}^{(k)}(\mathbf{x})$ . In particular, we have

$$\begin{aligned} (\mathcal{L}\mathbf{u})(\mathbf{x}) &= \left(1 - \frac{\tilde{w}_\epsilon(\mathbf{x}, \mathbf{x})}{W}\right) u_h(\mathbf{x}) \\ &\quad - \sum_{\mathbf{y} \in S_{\epsilon,h}^{(k)}(\mathbf{x}) \setminus \{\mathbf{x}\}} \frac{\tilde{w}_\epsilon(\mathbf{x}, \mathbf{y})}{W} u_h(\mathbf{y}), \end{aligned}$$

where  $W$  is the total mass and can be computed in one of two ways, using either the original weights  $w_\epsilon$  or the equivalent weights  $\tilde{w}_\epsilon$ , exploiting preservation of mass (10):

$$W := \sum_{\mathbf{y} \in S_{\epsilon,h}^{(k)}(\mathbf{x}) \cup (\text{Supp}(A_{\epsilon,h}(\mathbf{x})) \cap (\Omega_h \setminus D_h^{(k)}))} \tilde{w}_\epsilon(\mathbf{x}, \mathbf{y}) \quad (16)$$

$$= \sum_{\mathbf{y} \in (A_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\}) \cap (\Omega \setminus D^{(k)})} w_\epsilon(\mathbf{x}, \mathbf{y}). \quad (17)$$

Generally, (17) is more convenient to work with than (16), but (16) combined with the inherited non-degeneracy condition (13) tells us that

$$\sum_{\mathbf{y} \in S_{\epsilon,h}^{(k)}(\mathbf{x})} \tilde{w}_{\epsilon}(\mathbf{x}, \mathbf{y}) < W,$$

because the non-degeneracy conditions implies that a non-zero proportion of the total weight goes into the known pixels in  $\text{Supp}(A_{\epsilon,h}(\mathbf{x})) \cap (\Omega \setminus D_h^{(k)})$ . From this it immediately follows that  $\mathcal{L}$  is strictly diagonally dominant as

$$\begin{aligned} |\mathcal{L}_{\mathbf{x},\mathbf{x}}| - \sum_{\mathbf{y} \in \partial_{\text{ready}} D_h^{(k)} \setminus \{\mathbf{x}\}} |\mathcal{L}_{\mathbf{x},\mathbf{y}}| \\ &= \left(1 - \frac{\tilde{w}_{\epsilon}(\mathbf{x}, \mathbf{x})}{W}\right) - \sum_{\mathbf{y} \in S_{\epsilon,h}^{(k)}(\mathbf{x}) \setminus \{\mathbf{x}\}} \frac{\tilde{w}_{\epsilon}(\mathbf{x}, \mathbf{y})}{W} \\ &= 1 - \sum_{\mathbf{y} \in S_{\epsilon,h}^{(k)}(\mathbf{x})} \frac{\tilde{w}_{\epsilon}(\mathbf{x}, \mathbf{y})}{W} > 0. \end{aligned}$$

To compute  $\mathbf{f}$ , we do not need the concept of equivalent weights. We have

$$\mathbf{f}(\mathbf{x}) = \sum_{\mathbf{y} \in (A_{\epsilon,h}(\mathbf{x}) \setminus \{\mathbf{x}\}) \cap (\Omega \setminus D^{(k)})} \frac{w_{\epsilon}(\mathbf{x}, \mathbf{y})}{W} u_h(\mathbf{y}). \quad (18)$$

## 5.1 Solving the Linear System

Designing maximally efficient methods for solving (15) is beyond the scope of this paper—our main purpose lies in understanding the effect of this extension on the continuum limit that we will derive later. Therefore, in this paper we consider only two very simple methods: damped Jacobi and SOR (successive over-relaxation). These are natural choices for a number of reasons. First, since  $\mathcal{L}$  is strictly diagonally dominant, these methods are both guaranteed to converge [42, Theorem 3.10, p. 79], at least in the case  $\omega = 1$ , where they reduce to Jacobi and Gauss–Seidel. Second, at least for the semi-implicit extension of Guidicelli, the performance of SOR is already satisfactory, (see Sect. 6.2, Proposition 1). Third, both methods can be implemented with minimal changes to the direct form of Algorithm 1. In fact, changing the variable “semiImplicit” from “false” to “true” in Algorithm 1 and executing the “FillBoundary” subroutine in parallel is equivalent to solving (15) using damped Jacobi, with underrelaxation parameter

$$\omega^* = \left(1 - \frac{\tilde{w}_{\epsilon}(\mathbf{x}, \mathbf{x})}{W}\right) \leq 1. \quad (19)$$



Similarly, executing the “FillBoundary” subroutine sequentially results in SOR with the same underrelaxation parameter—see Proposition 3 in “Appendix C” for a proof. Note that  $\omega^*$  is typically very close 1, so these methods are very similar to plain Jacobi and Gauss–Seidel.

**Remark 8** The reason Algorithm 1 with “semiImplicit” set to “true” results in damped Jacobi/SOR, rather than plain Jacobi/Gauss–Seidel, is because even though the update formula (1) for the  $n$ th iterate  $u_h^{(n)}(\mathbf{x})$  is expressed as a sum of the  $(n - 1)$ st iterate evaluated at *ghost pixels* that do not include  $\mathbf{x}$ , some of those ghost pixels may indirectly depend on  $u_h^{(n-1)}(\mathbf{x})$  because they are defined using bilinear interpolation. The result is that the  $n$ th iterate  $u_h^{(n)}(\mathbf{x})$  depends on  $u_h^{(n-1)}(\mathbf{x})$ , which is true of damped Jacobi/SOR but not of Jacobi/Gauss–Seidel.

## 6 Analysis

This section contains our core analysis, firstly of the convergence properties of semi-implicit Guidefill, then of the convergence of both Algorithm 1 and its semi-implicit extension to an asymptotic limit where  $r = \epsilon/h$  is fixed and  $h > 0$  but  $h \ll 1$  as well as a continuum limit where  $(h, \epsilon) \rightarrow (0, 0)$  along the ray  $\epsilon = rh$  (with  $r \in \mathbb{N}$  fixed). Next, in Sect. 7, we will apply our results to explain some of the artifacts discussed in Sect. 2. We begin with a discussion of some symmetry assumptions that will hold from now on (Fig. 12).

### 6.1 Symmetry Assumptions

We will assume throughout that the inpainting domain  $D$  is the unit square  $[0, 1) \times (0, 1]$  while the image domain is  $\Omega = [0, 1) \times (-\delta, 1]$  equipped with Dirichlet or periodic boundary conditions at  $x = 0$  and  $x = 1$ , and no condition at  $y = 1$ . We denote the undamaged portion of the image by

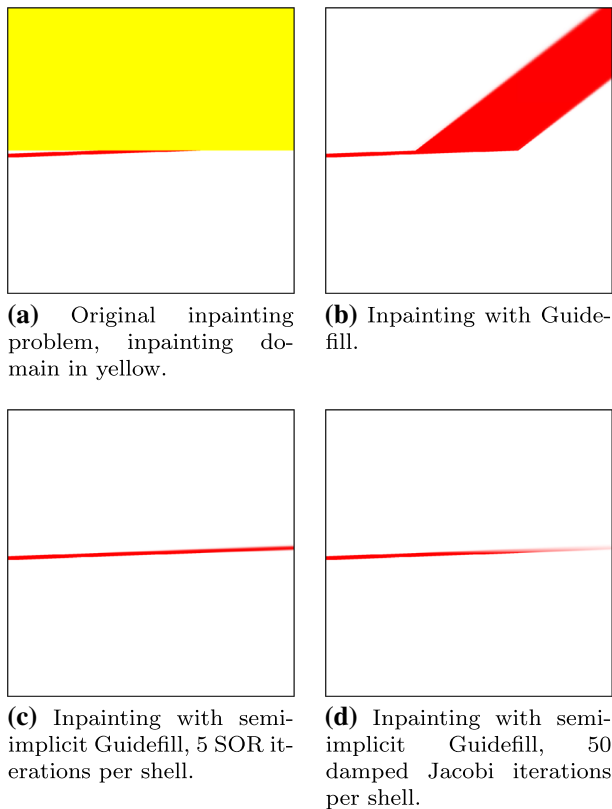
$$\mathcal{U} := [0, 1) \times (-\delta, 0] \quad \text{and} \quad \mathcal{U}_h := \mathcal{U} \cap \mathbb{Z}_h^2. \quad (20)$$

We discretize  $D = [0, 1) \times (0, 1]$  as an  $N \times N$  array of pixels  $D_h = D \cap (h \cdot \mathbb{Z}^2)$  with pixel width  $h := 1/N$ . Specifically, we have

$$D_h = \{(ih, jh) : 0 \leq i < N, 1 \leq j \leq N\}.$$

In order to ensure that the update formula (1) is well defined, we need  $\epsilon + 2h < \delta$ , which we achieve by assuming  $h < \frac{\delta}{r+2}$  [this follows from the inclusion (14)]. We assume that the default onion shell ordering is used, so that

$$\partial D_h^{(k)} = \{(jh, kh)\}_{j=1}^N.$$



**Fig. 12** Semi-implicit Guidefill and shallow inpainting Directions: In **a** a line makes a very shallow angle of just  $2^\circ$  with the inpainting domain, shown in yellow. This line is then inpainted using first Guidefill (**b**) and then the semi-implicit extension thereof (**c**). The latter uses 5 iterations of SOR per shell to solve the linear system (15) arising in every shell (in this case, the original image is  $2000 \times 2000$ px, so there are 1000 shells). Visually identical results can be obtained using damped Jacobi, but more than 100 iterations per shell are required—see Proposition 1. Both methods use relaxation parameter  $\omega = \omega^*$  (19) and are given  $\mathbf{g} = (\cos 2^\circ, \sin 2^\circ)$ ,  $\mu = 100$ ,  $\epsilon = 3$ px, and use the default onion shell ordering (smart-order is turned off). Guidefill kinks while its extension does not. In **d** we see the result of failing to solve the linear system (15) to sufficient accuracy by applying too few iterations of damped Jacobi. In this case only 50 iterations per shell are used and the extrapolated line gradually fades away

We also assume that the sets  $A_{\epsilon, h}(\mathbf{x})$  are translations of one another, and the weights  $w_\epsilon(\mathbf{x}, \mathbf{y})$  depend only on  $\frac{\mathbf{y} - \mathbf{x}}{\epsilon}$ , that is

$$w_\epsilon(\mathbf{x}, \mathbf{y}) = \hat{w}\left(\frac{\mathbf{y} - \mathbf{x}}{\epsilon}\right)$$

For coherence transport and Guidefill this means that the guidance direction  $\mathbf{g}$  is a constant.

**Remark 9** We make the above assumptions not because we believe they are necessary, but because they enable us to make our analysis as simple as possible while

still capturing the phenomena we would like to capture. In particular, the above two assumptions on the weights  $w_\epsilon$  and neighborhood  $A_{\epsilon,h}(\mathbf{x})$  ensure that the matrix  $\mathcal{L}$  given by (15) is either Toeplitz or circulant (depending on the boundary conditions), and also ensures that the random walk we connect Algorithm 1 to in Sect. 6.3 has i.i.d. (independent identically distributed) increments. Without these simplifications, our already lengthy analysis would become even more technical. Numerical experiments (omitted for brevity) suggest that these assumptions can be weakened, but proving this is beyond the scope of the present work.

These assumptions give us a high level of symmetry with which we may rewrite the update formula (1) of Algorithm 1 in the generic form

$$u_h(\mathbf{x}) = \frac{\sum_{\mathbf{y} \in a_r^*} w_r(\mathbf{0}, \mathbf{y}) u_h(\mathbf{x} + h\mathbf{y})}{\sum_{\mathbf{y} \in a_r^*} w_r(\mathbf{0}, \mathbf{y})} \quad (21)$$

where

$$a_r^* = \left( \frac{1}{h} A_{\epsilon,h}(\mathbf{0}) \setminus \{\mathbf{0}\} \right) \cap \{y \leq \delta\},$$

and  $\delta = -1$  for the direct method, while  $\delta = 0$  for the semi-implicit extension. In particular, for coherence transport we have  $a_r^* = b_r^-$  for the direct method and  $a_r^* = b_r^0$  for the extension, where

$$b_r^0 := \{(n, m) \in \mathbb{Z}^2 : 0 < n^2 + m^2 \leq r^2, m \leq 0\}. \quad (22)$$

$$b_r^- := \{(n, m) \in \mathbb{Z}^2 : n^2 + m^2 \leq r^2, m \leq -1\}. \quad (23)$$

Similarly, for Guidefill, we have  $a_r^* = \tilde{b}_r^-$  for the direct method and  $a_r^* = \tilde{b}_r^0$  for the semi-implicit extension, where

$$\tilde{b}_r^0 := \{n\hat{\mathbf{g}} + m\hat{\mathbf{g}}^\perp : (n, m) \in \mathbb{Z}^2, 0 < n^2 + m^2 \leq r^2,$$

$$n\hat{\mathbf{g}} \cdot \mathbf{e}_2 + m\hat{\mathbf{g}}^\perp \cdot \mathbf{e}_2 \leq 0\}$$

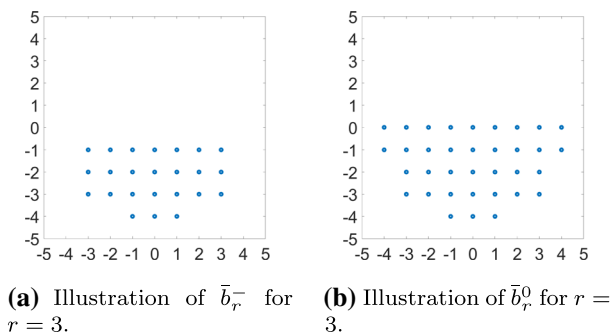
$$\tilde{b}_r^- := \{n\hat{\mathbf{g}} + m\hat{\mathbf{g}}^\perp : (n, m) \in \mathbb{Z}^2, n^2 + m^2 \leq r^2,$$

$$n\hat{\mathbf{g}} \cdot \mathbf{e}_2 + m\hat{\mathbf{g}}^\perp \cdot \mathbf{e}_2 \leq -1\},$$

and  $\hat{\mathbf{g}} := \mathbf{g}/\|\mathbf{g}\|$  (if  $\mathbf{g} = \mathbf{0}$  we set  $\tilde{b}_r^- = b_r^-$ ). The sets  $b_r^0$ ,  $\tilde{b}_r^-$ ,  $\tilde{b}_r^0$  may be visualized by looking at the portion of Fig. 10a, b on or below the lines  $y = 0$  and  $y = -1$  respectively. Also important are the dilated sets  $\bar{b}_r^0 = D(b_r^0) \cap \{y \leq 0\}$  and  $\bar{b}_r^- = D(b_r^-) \cap \{y \leq -1\}$ . These sets are given explicitly by

$$\begin{aligned} \bar{b}_r^0 &:= \{(n + \Delta n, m + \Delta m) : (n, m) \in b_r^0, (\Delta n, \Delta m) \\ &\quad \in \{-1, 0, 1\} \times \{-1, 0, 1\}, m + \Delta m \leq 0\} \end{aligned} \quad (24)$$

$$\begin{aligned} \bar{b}_r^- &:= \{(n + \Delta n, m + \Delta m) : (n, m) \in b_r^-, (\Delta n, \Delta m) \\ &\quad \in \{-1, 0, 1\} \times \{-1, 0, 1\}, m + \Delta m \leq -1\}. \end{aligned} \quad (25)$$



**Fig. 13** Visualization  $\bar{b}_r^-$  and  $\bar{b}_r^0$ : here we illustrate the sets  $\bar{b}_r^- := D(b_r^-) \cap \{y \leq -1\}$  and  $\bar{b}_r^0 := D(b_r^0) \cap \{y \leq 0\}$  in the case  $r = 3$ . These sets are defined explicitly in (23) and (24), and  $D$  is the dilation operator defined in the notation section. These sets are important because depending on whether we use the direct form of Algorithm 1 or its semi-implicit extension,  $\text{Supp}(a_r^*)$  is always contained in one or the other. See (26) in the text

This is because the universal support property (14) gives us the inclusion

$$\text{Supp}(a_r^*) \subseteq \begin{cases} \bar{b}_r^- \subseteq b_{r+2}^- & \text{direct form of Algorithm 1.} \\ \bar{b}_r^0 \subseteq b_{r+2}^0 & \text{semi-implicit form.} \end{cases} \quad (26)$$

which will be critical later. The sets  $\bar{b}_r^-$  and  $\bar{b}_r^0$  are illustrated in Fig. 13.

**Definition 1** We call the set  $a_r^*$  and the weights  $\{w_r(\mathbf{0}, \mathbf{y}) : \mathbf{y} \in a_r^*\}$  the *stencil* and *stencil weights* of a method. The *center of mass* of  $a_r^*$  is defined in the following two equivalent ways:

$$\text{C.M.} = \frac{\sum_{\mathbf{y} \in a_r^*} w_r(\mathbf{0}, \mathbf{y}) \mathbf{y}}{\sum_{\mathbf{y} \in a_r^*} w_r(\mathbf{0}, \mathbf{y})} = \frac{\sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \tilde{w}_r(\mathbf{0}, \mathbf{y}) \mathbf{y}}{\sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \tilde{w}_r(\mathbf{0}, \mathbf{y})}.$$

Here  $\tilde{w}_r$  denote the equivalent weights from Sect. 4, and the above identity follows from (10) and (11). The center of mass of  $a_r^*$  will play a critical role both in the continuum limit of Algorithm 1 (where it is the transport direction of the resulting transport PDE) and in the connection to random walks (where, after multiplication by  $h$ , it is the mean of the increments of the walk).

Under these assumptions, the matrix  $\mathcal{L}$  from (15) is independent of  $k$  (that is, we solve the same linear system for every shell), and moreover  $\mathcal{L}$  becomes a Toeplitz matrix (Dirichlet boundary conditions) or circulant matrix (periodic boundary conditions). For a given pixel  $\mathbf{x}$  at least  $r + 2$  pixels away from the boundary at  $x = 0$  and  $x = 1$ , that is  $\mathbf{x} \in \{(jh, kh) : r + 2 \leq j \leq N - r - 2\}$ , it takes on the form

$$(\mathcal{L}\mathbf{u})(\mathbf{x}) = \left(1 - \frac{\tilde{w}_r(\mathbf{0}, \mathbf{0})}{W}\right) u_h(\mathbf{x}) - \sum_{\mathbf{y} \in s_r \setminus \{\mathbf{0}\}} \frac{\tilde{w}_r(\mathbf{0}, \mathbf{y})}{W} u_h(\mathbf{x} + h\mathbf{y}),$$

where by (26) we have

$$\begin{aligned} s_r &= \text{Supp}(a_r^*) \cap (\mathbb{Z} \times \{0\}) \\ &\subseteq \{-(r+2)e_1, -(r+1)e_1, \dots, (r+1)e_1, (r+2)e_1\}. \end{aligned}$$

If  $\mathbf{x}$  is *not* at least  $r+2$  pixels away from the boundaries, then the formula changes in the usual way for Toeplitz and circulant matrices—we assume the reader is familiar with this and no further discussion is needed. Under the same assumptions the vector  $\mathbf{f}$  becomes

$$\mathbf{f} = \sum_{\mathbf{y} \in \text{Supp}(a_r^*) \setminus (\mathbb{Z} \times \{0\})} \frac{\tilde{w}_r(\mathbf{0}, \mathbf{y})}{W} u_h(\mathbf{x} + h\mathbf{y})$$

where  $\text{Supp}(a_r^*) \setminus (\mathbb{Z} \times \{0\}) \subseteq b_{r+2}^-$ . We also define

$$\tilde{W} := \sum_{j=-r-2}^{r+2} \tilde{w}_r(\mathbf{0}, je_1) \quad \text{and} \quad \tilde{w}_{0,0} := \tilde{w}_r(\mathbf{0}, \mathbf{0}). \quad (27)$$

For a given point  $\mathbf{x} \in \partial D_h^{(k)}$ , (again, assuming  $\mathbf{x}$  is far enough from the boundary) the ratio  $\frac{\tilde{W}}{W}$  gives the fraction of the mass of the stencil (Definition 1) centered at  $\mathbf{x}$  that gets “leaked” to the unknown pixels in  $\partial D_h^{(k)}$ , while  $\frac{\tilde{w}_{0,0}}{W}$  gives the fraction that gets leaked to  $\mathbf{x}$ . Together, these give a measure of the diagonal dominance of  $\mathcal{L}$ , as we have

$$\frac{\sum_{j \neq i} |\mathcal{L}_{ij}|}{|\mathcal{L}_{ii}|} = \frac{\tilde{W} - \tilde{w}_{0,0}}{W - \tilde{w}_{0,0}}.$$

The smaller this ratio is, the stronger the diagonal dominance of  $\mathcal{L}$ , and the faster damped Jacobi and SOR can be expected to converge—see Proposition 1 for explicit formulas.

For semi-implicit Guidefill with guidance direction  $\mathbf{g} = (\cos \theta, \sin \theta)$ , it can be shown that  $\mathcal{L}$  becomes a lower triangular matrix in the limit  $\mu \rightarrow \infty$ , provided we order unknowns left to right if  $\cos \theta > 0$  and right to left otherwise (see “Appendix D”). This gives us a hint that Gauss–Seidel and SOR might be very effective for the solution of (15) in this case, and indeed Proposition 1 confirms this.

## 6.2 Convergence Rates of Damped Jacobi and SOR for Semi-Implicit Guidefill

Here we derive tight bounds on the convergence rates of damped Jacobi and SOR for solving (15) in the semi-implicit extension of Guidefill described in Sect. 5, under the symmetry assumptions discussed above, and in the limit  $\mu \rightarrow \infty$  (recall that  $\mu$  is the parameter from the weights (6) controlling the extent to which weights are biased in favor of pixels in the directions  $\pm \mathbf{g}$ ). We will prove that in each case the

parameter value  $\omega = 1$  is optimal, but also pay special attention to the case  $\omega = \omega^*$  given by (19), since this is the value of  $\omega$  that our proposed implementation of the semi-implicit extension in Algorithm 1 uses. We consider general  $\omega$  mainly in order to demonstrate that the choice  $\omega = \omega^*$ , while not optimal, is close enough to optimal not to matter in practice.

We will assume that  $D = (0, 1]^2$  with Dirichlet boundary conditions, as this simplifies our analysis of SOR—for damped Jacobi, we could just as easily have assumed periodic boundary conditions. We will measure convergence rates with respect to the induced infinity norm, which obeys the identity

$$\|A\|_\infty = \max_{i=1}^N \sum_{j=1}^N |a_{ij}| \quad (28)$$

for any  $N \times N$  matrix  $A$ . Note that the iterates of the error  $\mathbf{e}^{(0)}, \mathbf{e}^{(1)}, \dots$  associated with any stationary iterative method with iteration matrix  $M$  obey the bounds

$$\|\mathbf{e}^{(n)}\| \leq \|M\|^n \|\mathbf{e}^{(0)}\| \text{ and } R(\mathbf{e}) := \sqrt[n]{\frac{\|\mathbf{e}^{(n)}\|}{\|\mathbf{e}^{(0)}\|}} \leq \|M\| \quad (29)$$

for any vector norm  $\|\cdot\|$  and induced matrix norm. We will be interested in these identities in the particular case that the vector norm is  $\|\cdot\|_\infty$ , and the stationary iterative method is damped Jacobi or SOR. Here

$$\mathbf{e}^{(n)} := u_h - u_h^{(n)}$$

denotes the difference between the exact solution to (15), found by first solving (15) to machine precision, with the approximate solution at iteration  $n$ .

**Proposition 1** *Suppose semi-implicit Guidefill with guidance direction  $\mathbf{g} = (\cos \theta, \sin \theta)$  is used to inpaint the square  $[0, 1]^2$  under the assumptions above, using either damped Jacobi or SOR to solve (15). Suppose that in the case of SOR,  $\partial D_h^{(k)} = \{(ih, kh)\}_{i=0}^{N-1}$  is ordered from left to right if  $\cos \theta \geq 0$  and from right to left otherwise. Let  $\mathcal{L}$  be as in (16) and define  $\mathcal{L} = D - L - U$ , where  $D$ ,  $-L$ , and  $-U$  are the diagonal, strictly lower triangular, and strictly upper triangular parts of  $\mathcal{L}$  respectively. Let  $J_\omega$  and  $G_\omega$  denote the iteration matrices of damped Jacobi and SOR respectively with relaxation parameter  $\omega$ , that is*

$$\begin{aligned} J_\omega &= I - \omega D^{-1} \mathcal{L} \\ G_\omega &= (I - \omega D^{-1} L)^{-1} ((1 - \omega)I + D^{-1} U). \end{aligned}$$

Let  $r = \epsilon/h$  and define  $\theta_c = \arcsin(1/r)$ . Define, for  $\theta_c \leq \theta \leq \pi - \theta_c$ ,  $j^* = \lfloor \frac{1}{\sin \theta} \rfloor \leq r$ . Let  $W$  be the total weight (17), let  $\tilde{W}$  and  $\tilde{w}_{0,0}$  be as in (27). Then, in the limit as  $\mu \rightarrow \infty$ , we have

$$W = \sum_{j=1}^r \frac{1}{j}, \quad \tilde{w}_{0,0} = (1 - \sin \theta)(1 - |\cos \theta|)$$

$$\tilde{W} = \begin{cases} \sum_{j=1}^r \frac{1}{j} - r \sin \theta & \text{if } \theta \in (0, \theta_c] \cup [\pi - \theta_c, \pi) \\ \sum_{j=1}^{j^*} \frac{1}{j} - j^* \sin \theta & \text{if } \theta \in (\theta_c, \pi - \theta_c) \end{cases}$$

and

$$\|J_\omega\|_\infty = |1 - \omega| + \omega \left( \frac{\tilde{W} - \tilde{w}_{0,0}}{W - \tilde{w}_{0,0}} \right) \quad \text{for } \omega \in (0, 2)$$

$$\|G_\omega\|_\infty = \frac{|1 - \omega|}{1 - \omega \frac{\tilde{W} - \tilde{w}_{0,0}}{W - \tilde{w}_{0,0}}} \quad \text{for } \omega \in (0, 1],$$

where  $\|\cdot\|_\infty$  is the induced infinity matrix norm (28). The optimal  $\omega \in (0, 2)$  is in both cases independent of  $\theta$  and equal to  $\omega = 1$ , where we obtain

$$\|J_1\|_\infty = 1 - \frac{r \sin \theta}{\sum_{j=1}^r \frac{1}{j} - (1 - \sin \theta)(1 - |\cos \theta|)}$$

if  $\theta \in (0, \theta_c] \cup [\pi - \theta_c, \pi)$ ,

$$\|J_1\|_\infty = 1 - \frac{\sum_{j=j^*+1}^r \frac{1}{j} + j^* \sin \theta}{\sum_{j=1}^r \frac{1}{j} - (1 - \sin \theta)(1 - |\cos \theta|)}$$

if  $\theta \in (\theta_c, \pi - \theta_c)$ , and

$$\|G_1\|_\infty = 0.$$

**Proof** “Appendix D”. □

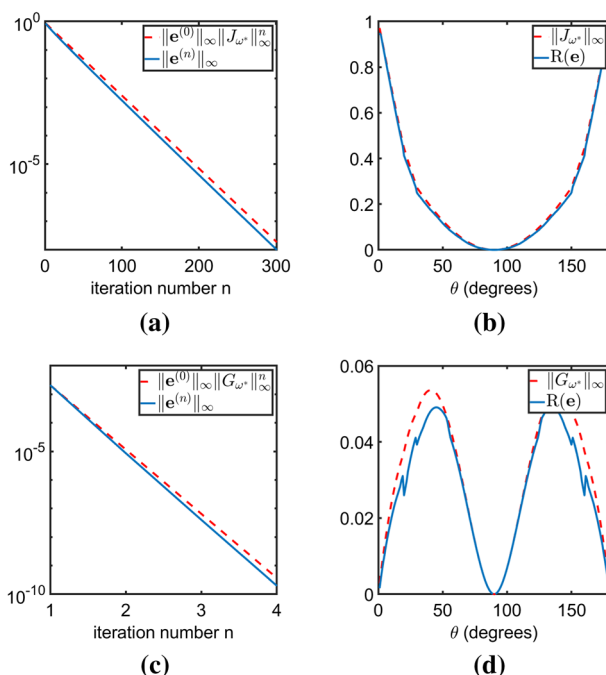
**Corollary 1** For the special case of

$$\omega^* = \left( 1 - \frac{\tilde{w}_{0,0}}{W} \right) = \left( 1 - \frac{(1 - \sin \theta)(1 - |\cos \theta|)}{\sum_{j=1}^r \frac{1}{j}} \right)$$

that is, the parameter value equivalent to running Algorithm 1 with “semiImplicit” set to true, we obtain

$$\|G_{\omega^*}\|_\infty = \frac{\tilde{w}_{0,0}}{W - \tilde{W}_k + \tilde{w}_{0,0}}$$

$$\|J_{\omega^*}\|_\infty = \frac{\tilde{W}_k}{W}$$



**Fig. 14** Convergence rates of damped Jacobi and SOR for semi-implicit Guidedfill: here we compare the experimentally measured convergence rates of the implementation of semi-implicit Guidedfill outlined in the blue text of Algorithm 1 ( $r = 3$ ,  $\mathbf{g} = (\cos \theta, \sin \theta)$  and  $\mu = 100$ ) with the theoretical bounds on  $\|J_{\omega^*}\|_{\infty}$  and  $\|G_{\omega^*}\|_{\infty}$  from Corollary 1. Note the excellent performance of SOR in comparison with damped Jacobi

or

$$\|G_{\omega^*}\|_{\infty} = \frac{(1 - \sin \theta)(1 - |\cos \theta|)}{r \sin \theta + (1 - \sin \theta)(1 - |\cos \theta|)}$$

$$\|J_{\omega^*}\|_{\infty} = 1 - \frac{r \sin \theta}{\sum_{j=1}^r \frac{1}{j}}$$

if  $\theta \in (0, \theta_c] \cup [\pi - \theta_c, \pi)$  and

$$\|G_{\omega^*}\|_{\infty} = \frac{(1 - \sin \theta)(1 - |\cos \theta|)}{\sum_{j=j^*+1}^r \frac{1}{j} + j^* \sin \theta + (1 - \sin \theta)(1 - |\cos \theta|)}$$

$$\|J_{\omega^*}\|_{\infty} = 1 - \frac{\sum_{j=j^*+1}^r \frac{1}{j} + j^* \sin \theta}{\sum_{j=1}^r \frac{1}{j}}$$

if  $\theta \in (\theta_c, \pi - \theta_c)$ . See Fig. 14 for a plot of  $\|G_{\omega^*}\|_{\infty}$  and  $\|J_{\omega^*}\|_{\infty}$  as a function of  $\theta$  for  $r = 3$ .

**Proof** This follows from direct substitution of  $\omega^*$  given by (19) into Proposition 1.  $\square$



As noted in Sect. 5, the implementation of semi-implicit Guidefill outlined in Algorithm 1 (blue text) is equivalent to solving the linear system (15) iteratively using damped Jacobi (parallel implementation) or SOR (sequential implementation), with relaxation parameter  $\omega^*$  given by (19). Figure 14 compares the experimentally measured convergence rates of the implementation of semi-implicit Guidefill outlined in the blue text of Algorithm 1 ( $r = 3$ ,  $\mathbf{g} = (\cos \theta, \sin \theta)$  and  $\mu = 100$ ) with the theoretical bounds on  $\|J_{\omega^*}\|_\infty$  and  $\|G_{\omega^*}\|_\infty$  from Corollary 1. Specifically, (a) and (c) confirm experimentally the first bound in (29) in the cases  $M = J_{\omega^*}$  and  $M = G_{\omega^*}$ , that is, damped Jacobi and SOR with relaxation parameter  $\omega^*$ , for the case  $\theta = 2^\circ$ . The inpainting problem in this case is the same as in Fig. 12a, and all the parameters of semi-implicit Guidefill are the same. The “exact” solution  $u_h$  was found by first solving (15) to machine precision. In each case, we measured convergence rates only within the first “shell” of the inpainting problem. Next, (b)–(d) confirm experimentally the second bound in (29), as a function of  $\theta$ . The inpainting problem is the same as the one in Fig. 5a, and all parameters are the same. In this case we vary  $\theta$  from  $1^\circ$  up to  $179^\circ$  in increments of one degree, in each case iteratively solving (15) (again, only for the first shell), computing  $R(\mathbf{e})$ , and comparing with  $\|J_{\omega^*}\|_\infty$  and  $\|G_{\omega^*}\|_\infty$ . Note the excellent performance of SOR in comparison with damped Jacobi.

Although our choice of  $\omega^*$  is non-optimal, it is convenient to implement and the difference in performance is negligible. In particular, for  $r = 3$  we have  $\|G_{\omega^*}\|_\infty \leq R(\mathbf{e}) \leq 0.06$  independent of  $\theta$ . Moreover, it follows from Corollary 1 that  $\|G_{\omega^*}\|_\infty$  is decreasing function of  $r$  for each fixed  $\theta$ , so this bound holds for all  $r \geq 3$  as well.

### 6.3 Convergence of Algorithm 1 to Continuum and Asymptotic Limits

In this section we prove the convergence of the direct and semi-implicit forms of Algorithm 1 to an “asymptotic limit” and a “fixed-ratio continuum limit”. Then, in Sect. 7, we use these limits to explain the kinking and blur artifacts of Algorithm 1 observed in practice as outlined in Sect. 2. We begin by defining what these limits are: **Definition of the continuum limit** We wish to prove convergence of the direct and semi-implicit forms of Algorithm 1 to a continuum limit  $u$  given by the transport equation

$$\nabla u \cdot \mathbf{g}_r^* = 0, \quad u|_{y=0} = u_0|_{y=0} \quad u|_{x=0} = u|_{x=1} \quad (30)$$

Specifically, we wish to prove convergence when we take  $(h, \epsilon) \rightarrow (0, 0)$  along the path  $\epsilon = rh$  (with  $r \in \mathbb{N}$  fixed). Because of the assumptions we have made in Sect. 6.1,  $\mathbf{g}_r^*$  will turn out to be a constant equal to the center of mass of the stencil  $a_r^*$  with respect to the stencil weights  $\{w_r(\mathbf{0}, \mathbf{y}) : \mathbf{y} \in a_r^*\}$  (Definition 1), that is

$$\mathbf{g}_r^* = \frac{\sum_{\mathbf{y} \in a_r^*} w_r(\mathbf{0}, \mathbf{y}) \mathbf{y}}{\sum_{\mathbf{y} \in a_r^*} w_r(\mathbf{0}, \mathbf{y})}. \quad (31)$$

As we will allow discontinuous boundary data  $u_0$ , the solution to (30) must be defined in a weak sense. However, since  $\mathbf{g}_r^*$  is a constant, this is simple. So long as  $\mathbf{g}_r^* \cdot \mathbf{e}_2 \neq 0$ ,

we simply define the solution to the transport problem (30) to be

$$u(\mathbf{x}) = u_0(\Pi_{\theta_r^*}(\mathbf{x})) \quad (32)$$

where

$$\Pi_{\theta_r^*}(x, y) = (x - \cot(\theta_r^*)y \mod 1, 0). \quad (33)$$

We call  $\Pi_{\theta_r^*} : D \rightarrow \partial D$  the transport operator associated with (32). The mod 1 is due to our assumed periodic boundary conditions and

$$\theta_r^* = \theta(\mathbf{g}_r^*) \in (0, \pi)$$

is the counterclockwise angle between the x-axis and the line  $L_{\mathbf{g}_r^*} := \{\lambda \mathbf{g}_r^* : \lambda \in \mathbb{R}\}$ .

**Remark 10** When  $u_0$  is smooth, convergence is straightforward to prove—we did so in [27, Theorem 1] for the direct form of Algorithm 1 using a simple argument based on Taylor series. However, in this paper we are interested in a more general setting where  $u_0$  may not be smooth and Taylor series may not be available. Instead of Taylor series, our main tool will be a connection to stopped random walks.

**Definition of the asymptotic limit** Unlike the continuum limit, which finds the limiting value of  $u_h$  when  $(h, \epsilon) \rightarrow (0, 0)$  along the path  $\epsilon = rh$  with  $r \in \mathbb{N}$  fixed, the asymptotic limit gives the asymptotic dependence of  $u_h$  on  $h$  when  $r = \epsilon/h$  is fixed and  $h \ll 1$ . For a fixed  $\mathbf{x} = (x, y) \in D_h$  the asymptotic limit is computed in the following two steps:

1. Given *discrete* boundary data  $u_0 : \mathcal{U}_h \rightarrow \mathbb{R}$ , generate new boundary data  $\tilde{u}_0 : \partial U_h \rightarrow \mathbb{R}$  generated from  $u_0$  by convolving it with a *discrete* convolution kernel

$$\tilde{g}_{\sigma(y,h)}(x_1, x_2) = g_{\sigma(y,h)}(x_1) \Delta(x_2),$$

where  $g_{\sigma(y,h)}$  is a one-dimensional Gaussian with variance dependent on both  $h$  and  $y$ , and  $\Delta$  is a function that “mixes” the top  $r$  rows of  $u_0$  by taking a convex combination with weights independent of both  $h$  and  $y$ .

2.  $u_h$  is now given by the solution to the same transport equation (30) as in the fixed ratio continuum limit, but with the boundary data  $u_0$  replaced by  $\tilde{u}_0 = \tilde{g}_{\sigma(y,h)} * u_0$ . That is,

$$u_h(x, y) = (\tilde{g}_{\sigma(y,h)} * u_0)(\Pi_{\theta_r^*}(x, y)) + o(1).$$

**Remark 11** By  $(\tilde{g}_{\sigma(y,h)} * u_0)(\mathbf{x})$  we mean the *discrete* circular convolution

$$(\tilde{g}_{\sigma(y,h)} * u_0)(\mathbf{x}) := \sum_{\mathbf{y} \in \mathcal{U}_h} u_0(\mathbf{y}) \tilde{g}_{\sigma(y,h)}(\mathbf{x} - \mathbf{y}),$$

where the  $x$ -coordinate of  $\mathbf{x} - \mathbf{y}$  is defined modulo 1. Strictly speaking, for this definition to coincide with the usual definition of discrete convolution given in, for example, [35,

Ch. 6], we must have  $\mathbf{x} - \mathbf{y} \in \mathcal{U}_h$ . In our case  $\mathbf{x}$  may lie between pixel centers, so we do not make this restriction.

## 6.4 Connection to Stopped Random Walks

The convergence of Algorithm 1 to both limits above is established based on a connection between Algorithm 1 and stopped random walks. The purpose of this section is to explain this connection.

To begin, Note that the update formula (21) gives a relationship between  $u_h(\mathbf{x})$  and its immediate neighbors in  $a_r^*$ , which for now we assume obeys  $a_r^* \subseteq b_r^-$  or  $a_r^* \subseteq b_r^0$  (if this is not the case we can apply the method of equivalent weights from Sect. 4). Now suppose we modify (21) iteratively by repeated application of the following rule: for each  $\mathbf{y} \in a_r^*$ , if  $\mathbf{x} + h\mathbf{y} \in D_h$ , replace  $u_h(\mathbf{x} + h\mathbf{y})$  in the RHS of (21) with the RHS of a version of (21) where the LHS is evaluated at  $\mathbf{x} + h\mathbf{y}$  (in other words, we are substituting (21) into itself). Otherwise, if  $\mathbf{x} + h\mathbf{y} \in \mathcal{U}_h$ , we are already in the undamaged portion of the image, and we may replace  $u_h(\mathbf{x} + h\mathbf{y})$  with  $u_0(\mathbf{x} + h\mathbf{y})$ . Repeat this procedure until  $u_h(\mathbf{x})$  is entirely expressed as a weighted sum of  $u_0|_{\mathcal{U}_h}$ , that is

$$u_h(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{U}_h} \rho(\mathbf{y}) u_0(\mathbf{y}), \quad (34)$$

for some as of yet unknown weights  $\rho$ . Denoting  $\mathbf{x} := (nh, mh)$ , then for the direct form of Algorithm 1 this procedure will terminate after  $m$  steps, as in this case (21) expresses  $u_h(\mathbf{x})$  in terms of neighbors at least  $h$  units below it. On the other hand, for the semi-implicit extension, (34) has to be interpreted as a limit.

This elimination procedure has a natural interpretation in terms of stopped random walks. Since the weights  $\{\frac{w(\mathbf{0}, \mathbf{y})}{W}\}_{\mathbf{y} \in a_r^*}$  are non-negative and sum to 1, we can interpret them as the density of a two dimensional random vector  $\mathbf{Z} := (U, V)$  taking values in  $b_r^-$  or  $b_r^0$  with density

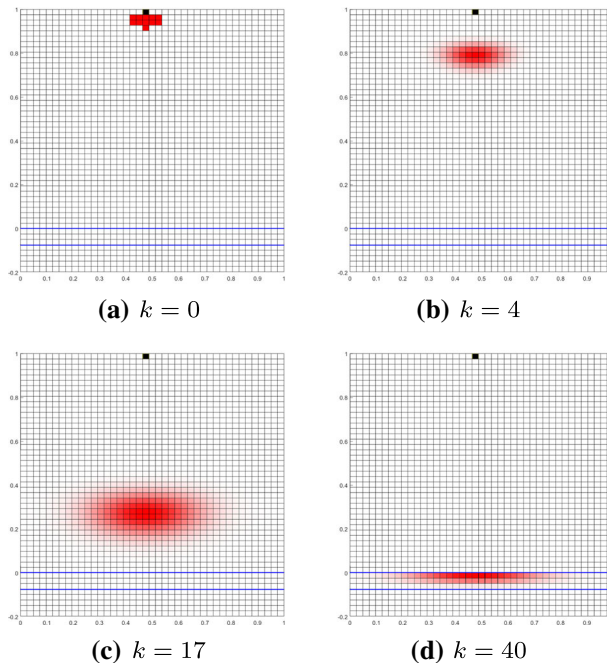
$$P(\mathbf{Z}_i = \mathbf{y}) = \frac{w_r(\mathbf{0}, \mathbf{y})}{W}. \quad (35)$$

Moreover, defining the random walk

$$\mathbf{X}_j := (X_j, Y_j) = (nh, mh) + h \sum_{i=1}^j \mathbf{Z}_i \quad (36)$$

with  $\{\mathbf{Z}_i\}$  i.i.d. and equal to  $\mathbf{Z}$  in distribution, and defining

$$\tau = \inf\{j : \mathbf{X}_j \in \mathcal{U}_h\} = \inf\{j : Y_j \leq 0\}, \quad (37)$$



**Fig. 15** Connection to stopped random walks: here we illustrate the connection between the elimination procedure described in the text and stopped random walks. In **a** the pixel  $\mathbf{x}$  (colored black) is expressed as a weighted average (1) of its neighbors in  $b_r^-$  (colored in red). In this case we use the direct form of Algorithm 1 with  $r = 3$  and uniform weights (which is why each neighbor in  $b_r^-$  is the same shade of red). In **c**, **d** we have applied  $k = 4$ ,  $k = 17$ , and  $k = 40$  steps of our elimination procedure (which essentially consists of substituting (1) into itself repeatedly—details in the text), and  $u_h(\mathbf{x})$  is now expressed as weighted sum of  $u_h(\mathbf{y})$  for whichever pixels  $\mathbf{y}$  are colored red, with darker shades of red indicating greater weight. The purpose of this procedure is to express the color of a given pixel  $\mathbf{x}$  deep inside the inpainting domain entirely in terms of the colors of known pixels below the line  $y = 0$  (shown in blue)

then after  $k$  steps of elimination, we have

$$u_h(\mathbf{x}) = \sum_{\mathbf{y} \in \Omega_h} \rho_{\mathbf{X}_{j \wedge \tau}}(\mathbf{y}) u_h(\mathbf{y}),$$

where  $\rho_{\mathbf{X}_{j \wedge \tau}}$  denotes the density of  $\mathbf{X}_{j \wedge \tau}$ . Denoting the mean of  $\mathbf{Z}$  by  $(\mu_x, \mu_y)$  note that by (31) we have the equivalence

$$(\mu_x, \mu_y) = \mathbf{g}_r^*.$$

In other words, the mean of  $\mathbf{Z}$  is precisely the transport direction of our limiting Eq. (30). The condition  $\mathbf{g}_r^* \cdot \mathbf{e}_2 \neq 0$ , which we needed for (30) to be defined, implies  $\mu_y < 0$ . In the nomenclature of random walks, this means that  $\mathbf{X}_k$  has negative drift in the  $y$  direction, while  $\tau$  is the first passage time through  $y = 0$ . Fortunately, this type of random walk and this type of first passage time have been studied and are well understood. See for example [24, Chapter 4], [22,23,25]. The book [24] also

provides an good overview of stopped random walks in general. In particular, we know immediately that  $\tau$  is a stopping time,  $P(\tau = \infty) = 0$ , and  $\tau$  has finite moments of all orders [24, Chapter 3, Theorems 1.1 and 3.1]. It follows that

$$u_h(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{L}_h} \rho_{\mathbf{X}_\tau}(\mathbf{y}) u_0(\mathbf{y}) = E[u_0(\mathbf{X}_\tau)]. \quad (38)$$

See Fig. 15 for an illustration of these ideas.

### Lattice coordinates and absolute coordinates

Our derivation of the asymptotic limit leverages known results for stopped random walks. However, the results most relevant to us concern random walks on the integer lattice  $\mathbb{Z}^2$ , whereas our random walk is on the scaled lattice  $h \cdot \mathbb{Z}^2$ . In order to facilitate conversion between our setting and the setting of the results we quote, we make the following definition

**Definition 2** Given  $\mathbf{x} = (x, y) = (ih, jh) \in D_h \subset h \cdot \mathbb{Z}^2$ , we refer to  $(i, j) \in \mathbb{Z}^2$  as the *lattice coordinates* of  $\mathbf{x}$ , while we refer to  $(x, y) = (ih, jh)$  as the *absolute coordinates* of  $\mathbf{x}$ .

### Overshoot and the ladder process associated with $\mathbf{X}_\tau$

In general, we expect that the stopped random walk  $\mathbf{X}_\tau$  will overshoot the line  $y = 0$ . The (asymptotic) nature of the overshoot probability density is well understood and is described briefly here (we will need it in Sect. 6.5). However, in order to describe it, we first need to define the *ladder process* associated with  $\mathbf{X}_\tau$ . The strict ladder process  $\{n_i, S_{n_i}\}_{i \geq 1}$  associated with the  $y$ -coordinate of  $\mathbf{X}_\tau$  is given by

$$S_n = \sum_{i=1}^n V_i, \quad (39)$$

$$n_1 = \inf\{n : V_n < 0\} \quad (40)$$

$$n_{i+1} = \inf\{n : S_n < S_{n_i}\}. \quad (41)$$

Following the notation in [36], we denote by  $H$  the law of  $V_{n_1}$ , and write  $v_y = E[V_{n_1}]$ . The (asymptotic) overshoot function  $\Delta$  is then given by the following identity [36, Equation (1.4)]:

$$\Delta(kh) = H((-\infty, k))/v_y. \quad (42)$$

in absolute coordinates or

$$\Delta(k) = H((-\infty, k))/v_y. \quad (43)$$

in lattice coordinates (Definition 2). Note that  $H$  here is a probability measure, so that  $H((-\infty, k))$  refers to the *measure* of the interval  $(-\infty, k)$ .

## 6.5 Derivation of the Asymptotic Limit

The key idea of our asymptotic limit is to combine (38) with known results on the asymptotic distribution of  $\rho_{\mathbf{X}_\tau}$  started from  $(x, y) = (nh, mh) \in D_h$  when  $m$  is very large (for fixed  $y$ , this is equivalent to  $h$  very small). This problem has been considered by a number of authors, including [24, Chapter 4], [22,23,25], where it is shown that  $\rho_{\mathbf{X}_\tau}$  obeys a central limit theorem, asymptotically approaching a normal distribution. However, for our purposes it is not enough to know the asymptotic distribution of the stopped walk for  $m \gg 1$ , we must also know the rate of convergence as a function of  $m$ . While many authors have considered the former, to our knowledge only [30,36] have considered the latter. Our results here are based on [36], specifically the corollary immediately following Theorem 3.3. The following theorem restates this result, along with a modified version adapted to our periodic boundary conditions.

**Theorem 1** Let  $\mathbf{X}_\tau := (X_\tau, Y_\tau) \in \mathbb{Z}^2$  be a stopped random walk started from  $(n, m) \in \mathbb{Z}^2$  and with i.i.d. increments  $\Delta \mathbf{X}_i = \mathbf{Z}_i$ . Assume that  $\mathbf{X}_\tau$  has negative drift, that is  $(\mu_x, \mu_y) = E[\mathbf{Z}_i]$  obeys  $\mu_y < 0$  and define the stopping time  $\tau$  by

$$\tau := \inf\{n : Y_n < 0\}.$$

Define the characteristic function  $\varphi_{\mathbf{Z}_i}$  of  $\mathbf{Z}_i$  by

$$\varphi_{\mathbf{Z}_i}(\mathbf{u}) = E[e^{\sqrt{-1}\mathbf{Z}_i \cdot \mathbf{u}}]$$

and assume that  $|\varphi_{\mathbf{Z}_i}| < 1$  unless each component of  $\mathbf{u}$  is an integer multiple of  $2\pi$ . Then

$$\lim_{m \rightarrow \infty} \sqrt{m} |\rho_{\mathbf{X}_\tau}(i, j) - \tilde{G}_{\sigma(m), \mu}(i, j)| = 0 \quad (44)$$

uniformly in  $(i, j) \in \mathbb{Z}^2$ , where

$$\begin{aligned} \tilde{G}_{\sigma(m), \mu}(i, j) &= G_{\sigma(m), \mu}(i) \Delta(j) \\ G_{\sigma(m), \mu}(i) &= \frac{e^{-\frac{(i-\mu)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} \end{aligned}$$

with

$$\sigma(m)^2 = \frac{\gamma^2 m}{|\mu_y|^3} \quad \text{where } \gamma^2 = \text{Var}(\mu_x V_1 - \mu_y U_1). \quad (45)$$

and

$$\mu = n - \frac{\mu_x}{\mu_y} m$$

and  $\Delta(j)$  is given by (43). Moreover, if the domain of  $\mathbf{X}_\tau$  is changed to

$$\mathbf{X}_\tau \in \{0, 1, \dots, N-1\} \times \{-(r+2), -(r+1), \dots, N\}$$

with periodic boundary condition  $0 \sim N$  in the  $x$ -direction, and if  $m \leq N$  while  $\mathbf{Z}_i := (U_i, V_i)$  obeys  $V_i \leq 0$ , then the same result holds but with  $G_{\sigma(m), \mu}(i)$  replaced by

$$G_{\sigma(m), \mu, N}^*(i) = \frac{e^{-\frac{d_N^{\text{circ}}(i, \mu^*)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)}$$

where

$$\mu^* = n - \frac{\mu_x}{\mu_y} m \mod N.$$

and  $d_N^{\text{circ}}$  denotes the circular distance defined for  $x, y \in \mathbb{R}$  by

$$d_N^{\text{circ}}(x, y) = \min(x - y \mod N, y - x \mod N) \quad (46)$$

If  $\mu = 0$ , we write  $G_\sigma$  and  $G_{\sigma, N}^*$  in place of  $G_{\sigma, 0}$  and  $G_{\sigma, 0, N}^*$ .

**Proof** The first statement follows from the (un-numbered) Corollary immediately following Theorem 3.3 in [36]. The second statement is a simple corollary, the details of which may be found in “Appendix F”.  $\square$

The asymptotic limit follows straightforwardly from Theorem 1 and (38) once we have completed the following tasks:

1. The second statement of Theorem 1 gives the asymptotic probability that  $X_\tau = i$  for every  $i = 0, \dots, N-1$  for our random walk with periodic boundary conditions. However, the Gaussian  $G_{\sigma(m), \mu, N}^*$  that  $X_\tau$  converges to is not normalized with respect to summation over  $\{i\}_{i=0}^{N-1}$ . Lemma 1 addresses this problem by allowing us to convert statements about  $G_{\sigma(m), N}^*$  into statements about  $g_{\sigma(m), N}^*$ , a discrete Gaussian that is normalized with respect to summation over the set  $\{i\}_{i=0}^{N-1}$ .
2. The assumption in Theorem 1 on the characteristic function of  $\mathbf{Z}_i$  can be inconvenient to check. Definition 3 gives an equivalent geometric condition (which can typically be verified by inspection) that is proven in Lemma 2 to be equivalent.
3. Theorem 1 is expressed with respect to the integer lattice  $\mathbb{Z}^2$ , whereas our random walk is on the scaled lattice  $h \cdot \mathbb{Z}^2$ . Lemma 3 uses Theorem 1 combined with Lemmas 1 and 2 to prove a version of Theorem 1 applicable to our setting.
4. We need to derive an expression for  $\Delta(x_2)$ . This is done in Lemma 4.
5. We need to derive a bound on the probability of  $\mathbf{X}_\tau$  deviating significantly from  $\Pi_{\theta^*}(\mathbf{x})$ . This is done in Lemma 5.

Once the above items are taken care of, we prove convergence to the asymptotic limit in Theorem 2.

We begin with task one. For now, we work in lattice coordinates (Definition 2). Let us denote by  $G_{\sigma,\mu}$  and  $G_{\sigma,\mu,N}^*$  the standard and periodic Gaussians with mean  $\mu$  and variance  $\sigma^2$  defined for  $x \in \mathbb{R}$  and  $x \in [0, N)$  respectively by

$$G_{\sigma,\mu}(x) = \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma} \quad G_{\sigma,\mu,N}^*(x) = \frac{e^{-\frac{d_N^{\text{circ}}(x,\mu)^2}{2\sigma^2}}}{\sqrt{2\pi}\sigma}. \quad (47)$$

These are the functions that  $X_\tau$  converges to in the two forms of Theorem 1 (infinite domain and periodic domain). However, neither of them are probability densities. This is because while

$$\int_{\mathbb{R}} G_{\sigma,\mu}(x) dx = 1$$

is true, the domain of  $X_\tau$  in the infinite case is not  $\mathbb{R}$ , but rather  $\mathbb{Z}$ , and

$$\sum_{i \in \mathbb{Z}} G_{\sigma,\mu}(i) \neq 1.$$

For  $G_{\sigma,\mu,N}^*$ , the situation is worse: we have

$$\int_0^N G_{\sigma,\mu,N}^*(x) dx \neq 1 \quad \text{and} \quad \sum_{i=0}^{N-1} G_{\sigma,\mu,N}^*(i) \neq 1.$$

We therefore consider it more natural to work with the discrete Gaussians

$$\begin{aligned} g_{\sigma,\mu}(i) &= \frac{e^{-\frac{(i-\mu)^2}{2\sigma^2}}}{\sum_{k \in \mathbb{Z}} e^{-\frac{(k-\mu)^2}{2\sigma^2}}} \\ g_{\sigma,\mu,N}^*(i) &= \frac{e^{-\frac{d_N^{\text{circ}}(i,\mu)^2}{2\sigma^2}}}{\sum_{k=0}^{N-1} e^{-\frac{d_N^{\text{circ}}(k,\mu)^2}{2\sigma^2}}} \end{aligned} \quad (48)$$

which are normalized by construction (similarly to  $G_{\sigma,\mu}$  and  $G_{\sigma,\mu,N}^*$ , when  $\mu = 0$  we simply write  $g_\sigma$  and  $g_{\sigma,N}^*$ ). The following lemma allows us to convert statements about  $G_{\sigma,\mu,N}^*$  into statements about  $g_{\sigma,\mu,N}^*$ . A nearly identical argument would allow us to do the same for  $G_{\sigma,\mu}$  and  $g_{\sigma,\mu}$ , but we do not pursue this here. Lemma 1 also contains a result regarding discrete sums of Gaussians which is to the best of our knowledge novel and of interest in and of itself.



**Lemma 1** *i. There is a constant  $C > 0$  such that for  $|\sigma| \geq 1$  we have*

$$\left| \sum_{n \in \mathbb{Z}} e^{-\frac{(n-\mu)^2}{2\sigma^2}} - \sqrt{2\pi}\sigma \right| \leq C\sigma e^{-2\pi^2\sigma^2} \quad (49)$$

*ii. Let  $G_{\sigma, N+1}^*$  and  $g_{\sigma, N+1}^*$  be defined as above, with common mean  $\mu$  and variance  $\sigma^2$ . Further suppose that  $\sigma^2$  is proportional to  $m \in \mathbb{N}$ , that is  $\sigma^2(m) = \tilde{\sigma}^2 m$  for some constant  $\tilde{\sigma}^2 > 0$ . Suppose further that  $m \leq N$  (recall that  $N = 1/h$ ). Then there exist constants  $A > 0$  and  $b > 0$  such that*

$$|G_{\sigma(m), N}^*(x) - g_{\sigma(m), N}^*(x)| < Am^{\frac{1}{2}} e^{-bm}$$

*uniformly in  $x \in \mathbb{R}$  for  $m$  sufficiently large.*

**Proof** We prove only the first statement. The second statement is then derived from the first together with routine tail estimates on gaussians, and is provided in “Appendix G”.

First we note that our sum is closely related to Jacobi theta functions of the third type, defined by [5]

$$\vartheta_3(q) = \sum_{n \in \mathbb{Z}} q^{n^2}.$$

This function is known explicitly for a few values of  $q$ , for example

$$\vartheta_3(e^{-\pi}) = \frac{\pi^{\frac{1}{4}}}{\Gamma(\frac{3}{4})}. \quad (50)$$

We will use this identity later on.

Our argument is based on the Poisson summation formula [2, pp. 322–333], which states that if  $f : \mathbb{R} \rightarrow \mathbb{R}$  is a Schwartz function (that is, decays faster than  $|x|^{-N}$  for all  $N \in \mathbb{N}$  for  $x$  sufficiently large), then

$$\sum_{n \in \mathbb{Z}} f(n) = \sum_{n \in \mathbb{Z}} \int_{-\infty}^{\infty} f(x) e^{-i2\pi nx} dx.$$

Choosing  $f(x) = e^{-\frac{(x-\mu)^2}{2\sigma^2}}$  gives

$$\begin{aligned} \sum_{n \in \mathbb{Z}} e^{-\frac{(n-\mu)^2}{2\sigma^2}} &= \sum_{n \in \mathbb{Z}} \int_{-\infty}^{\infty} e^{-\frac{(x-\mu)^2}{2\sigma^2}} e^{-i2\pi nx} dx \\ &= \sum_{n \in \mathbb{Z}} \int_{-\infty}^{\infty} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \cos(2\pi nx) dx, \end{aligned}$$

where on line two, noting that the LHS side is real, we have eliminated the imaginary part of the RHS as well. Changing variables in the integral to  $y = x - \mu$  gives

$$\begin{aligned}\sum_{n \in \mathbb{Z}} e^{-\frac{(n-\mu)^2}{2\sigma^2}} &= \sum_{n \in \mathbb{Z}} \int_{-\infty}^{\infty} e^{-\frac{y^2}{2\sigma^2}} \cos(2\pi ny + 2\pi n\mu) dy \\ &= \sum_{n \in \mathbb{Z}} \cos(2\pi n\mu) \int_{-\infty}^{\infty} e^{-\frac{y^2}{2\sigma^2}} \cos(2\pi ny) dy\end{aligned}$$

Since the sin terms in the expansion of  $\cos(2\pi nx + 2\pi n\mu)$  are odd functions, they integrate to 0, and we have eliminated them in line two. But it is well known that

$$\int_{-\infty}^{\infty} e^{-\frac{y^2}{2\sigma^2}} \cos(2\pi ny) dy = \sqrt{2\pi} \sigma e^{-2\pi^2 \sigma^2 n^2}$$

(this follows from the identity for the Fourier Transform of a Gaussian [8]). Hence

$$\sum_{n \in \mathbb{Z}} e^{-\frac{(n-\mu)^2}{2\sigma^2}} = \sqrt{2\pi} \sigma \sum_{n \in \mathbb{Z}} e^{-2\pi^2 \sigma^2 n^2} \cos(2\pi n\mu)$$

and therefore (using the fact that  $|\sigma| \geq 1$  on line four):

$$\begin{aligned}\left| \sum_{n \in \mathbb{Z}} e^{-\frac{(n-\mu)^2}{2\sigma^2}} - \sqrt{2\pi} \sigma \right| &\leq 2\sqrt{2\pi} \sum_{n=1}^{\infty} e^{-2\pi^2 \sigma^2 n^2} |\cos(2\pi n\mu)| \\ &\leq 2\sqrt{2\pi} \sum_{n=1}^{\infty} e^{-2\pi^2 \sigma^2 n^2} \\ &= 2\sqrt{2\pi} \sigma e^{-\pi^2 \sigma^2} \sum_{n=1}^{\infty} e^{-2\pi^2 \sigma^2 (n^2-1)} \\ &\leq 2\sqrt{2\pi} \sigma e^{-\pi^2 \sigma^2} \sum_{n=1}^{\infty} e^{-\pi(n^2-1)} \\ &= \sqrt{2\pi} e^{\pi} (\vartheta_3(e^{-\pi}) - 1) \sigma e^{-\pi^2 \sigma^2}.\end{aligned}$$

Using (50), claim i. then follows with

$$\begin{aligned}C &= \sqrt{2\pi} e^{\pi} (\vartheta_3(e^{-\pi}) - 1) \\ &= \sqrt{2\pi} e^{\pi} \left( \frac{\pi^{\frac{1}{4}}}{\Gamma(3/4)} - 1 \right) \approx 5.014.\end{aligned}$$

□

**Remark 12** Although statement i. of Lemma 1 is stated asymptotically, in practice the convergence is so rapid that we have

$$\sum_{n \in \mathbb{Z}} e^{-\frac{n^2}{2\sigma^2}} = \sqrt{2\pi} \sigma$$

to machine precise for  $|\sigma| > 2$ . This has been noted empirically by other authors, e.g. [20, p. 288], but as far as we know (49) is the first explicit bound. Note also that our bound (49) is extremely tight. For  $\sigma = 1$ , the first value of  $\sigma$  for which our bound is valid, the left hand side and right hand side of (49) differ by  $O(10^{-12})$ . For  $\alpha > 1$ , the bound rapidly becomes even tighter.

Having established a connection between discrete and continuous Gaussians, we now move onto task number two and derive a necessary and sufficient condition for the characteristic function  $\varphi_{\mathbf{Z}_i}$  of  $\mathbf{Z}_i$  to satisfy the conditions of Theorem 1.

**Definition 3** We say that a stencil  $a_r^*$  is “confined to a sublattice of  $\mathbb{Z}^2$ ” if there exists an  $(n, m) \in \mathbb{Z}^2$  and  $k_1, k_2 \in \mathbb{Z}$  with at least one of  $k_1, k_2$  greater than one in magnitude, such that the density of  $\text{Supp}(a_r^*)$  is zero outside of  $(n, m) + k_1\mathbb{Z} \times k_2\mathbb{Z}$ . We say that  $a_r^*$  is not confined to any sublattice of  $\mathbb{Z}^2$  if no such  $(n, m)$  and  $k_1, k_2$  exist.

**Lemma 2** The characteristic function of  $\mathbf{Z}_i$ , defined by

$$\varphi_{\mathbf{Z}_i}(\mathbf{u}) = E[e^{\sqrt{-1}\mathbf{Z}_i \cdot \mathbf{u}}]$$

obeys  $|\varphi_{\mathbf{Z}_i}| < 1$  unless each component of  $\mathbf{u}$  is an integer multiple of  $2\pi$  if and only if the associated stencil  $a_r^*$  is not confined to any sublattice of  $\mathbb{Z}^2$  (in the sense of Definition 3).

**Proof** First assume that  $a_r^*$  is not confined to any sublattice of  $\mathbb{Z}^2$ . For convenience, let us write  $\tilde{w}_{\mathbf{y}} := \tilde{w}_r(\mathbf{0}, \mathbf{y})$ . By the definition of expectation we have

$$\varphi_{\mathbf{Z}_i}(\mathbf{u}) = E[e^{\sqrt{-1}\mathbf{Z}_i \cdot \mathbf{u}}] = \frac{\sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \tilde{w}_{\mathbf{y}} e^{\sqrt{-1}\mathbf{y} \cdot \mathbf{u}}}{W}.$$

Since  $W \equiv \sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \tilde{w}_{\mathbf{y}}$  we have  $|\varphi_{\mathbf{Z}_i}(\mathbf{u})| \leq 1$ , with equality if and only if  $e^{\sqrt{-1}\mathbf{y} \cdot \mathbf{u}}$  have the same phase for all  $\mathbf{y} \in \text{Supp}(a_r^*)$  such that  $\tilde{w}_{\mathbf{y}} > 0$ . Since  $a_r^*$  is not confined to any sublattice of  $\mathbb{Z}^2$ , there must exist a  $\mathbf{y}_1 \in \text{Supp}(a_r^*)$  such that  $\tilde{w}_{\mathbf{y}_1} > 0$  and  $\tilde{w}_{\mathbf{y}_1 + \mathbf{e}_1} > 0$ . Thus

$$\begin{aligned} |\varphi_{\mathbf{Z}_i}(\mathbf{u})| = 1 &\Rightarrow e^{\sqrt{-1}\mathbf{y}_1 \cdot \mathbf{u}} = e^{\sqrt{-1}(\mathbf{y}_1 + \mathbf{e}_1) \cdot \mathbf{u}} \\ &\Rightarrow e^{\sqrt{-1}\mathbf{u} \cdot \mathbf{e}_1} = 1 \end{aligned}$$

after canceling a common factor of  $e^{\sqrt{-1}\mathbf{y}_1 \cdot \mathbf{u}}$ . But then  $\mathbf{u} \cdot \mathbf{e}_1$  must be an integer multiple of  $2\pi$ , and a similar argument with proves that the same must be true of  $\mathbf{u} \cdot \mathbf{e}_2$ .

Now assume that  $a_r^*$  is confined to some sublattice  $(n, m) + k_1\mathbb{Z} \times k_2\mathbb{Z}$  of  $\mathbb{Z}^2$ . Then defining  $\mathbf{u} := (u_1, u_2)$  and  $\mathbf{y} = (y_1, y_2)$  we can write

$$\begin{aligned} |\varphi_{\mathbf{Z}_i}(\mathbf{u})| &= \left| \sum_{\mathbf{y} \in \mathbb{Z}^2} \frac{\tilde{w}_{\mathbf{y}}}{W} e^{\sqrt{-1}(n+y_1k_1, m+y_2k_2) \cdot \mathbf{u}} \right| \\ &= \left| e^{\sqrt{-1}(n, m) \cdot \mathbf{u}} \sum_{\mathbf{y} \in \mathbb{Z}^2} \frac{\tilde{w}_{\mathbf{y}}}{W} e^{\sqrt{-1}(y_1k_1, y_2k_2) \cdot \mathbf{u}} \right| \\ &= \left| \sum_{\mathbf{y} \in \mathbb{Z}^2} \frac{\tilde{w}_{\mathbf{y}}}{W} e^{\sqrt{-1}(y_1k_1, y_2k_2) \cdot \mathbf{u}} \right| \end{aligned}$$

If  $u_1 = 2\pi/k_1$  and  $u_2 = 2\pi/k_2$ , then we have

$$|\varphi_{\mathbf{Z}_i}(\mathbf{u})| = \sum_{\mathbf{y} \in \mathbb{Z}^2} \frac{\tilde{w}_{\mathbf{y}}}{W} = 1.$$

Since at least one of  $k_1, k_2$  is greater than one in magnitude, at least one of  $u_1, u_2$  is not an integer multiple of  $2\pi$ .  $\square$

Next, we use the above lemma together with Lemma 1 to prove a version of Theorem 1 expressed in terms of a discrete Gaussian  $g_{\sigma(y, h), \mu^*, 1}^*$  rather than  $G_{\sigma(m), \mu^*, N}^*$ .

**Lemma 3** *Let  $a_r^*$  be a stencil not confined to any sublattice of  $\mathbb{Z}^2$  in the sense of Definition 3, or equivalently that the characteristic function of  $\mathbf{Z}_i$ , that is*

$$\varphi_{\mathbf{Z}_i}(\mathbf{u}) = \sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \frac{w_r(\mathbf{0}, \mathbf{y}) e^{\sqrt{-1}\mathbf{y} \cdot \mathbf{u}}}{W},$$

*obeys  $|\varphi_{\mathbf{Z}_i}(\mathbf{u})| < 1$  unless both components of  $\mathbf{u}$  are integer multiples of  $2\pi$ . Let  $\mathbf{x} = (x, y) \in D_h$ , and let  $\mathbf{X}_\tau$  denote the stopped random walk started from  $\mathbf{x}$  described in Sect. 6.4. Then for  $y$  fixed, we have*

$$\rho_{X_\tau}(ih, jh) = \tilde{g}_{\sigma(y, h), \hat{x}}(ih, jh) + o(\sqrt{h})$$

where

$$\tilde{g}_{\sigma(y, h), \hat{x}}(ih, jh) = g_{\sigma(y, h), \hat{x}, 1}^*(ih) \Delta(jh)$$

with  $\Delta(jh)$  given by (42) and where  $g_{\sigma(y, h), \hat{x}, 1}^*$  is the one dimensional discrete periodic gaussian (48) with mean

$$\hat{x} = x - \frac{\mu_x}{\mu_y} y \pmod{1} \equiv \Pi_{\theta^*}(x, y),$$

where  $\Pi_{\theta_r^*}$  is the transport operator (33), and with variance

$$\sigma(y, h)^2 = \frac{\gamma^2 y h}{|\mu_y|^3} \quad \text{where } \gamma^2 = \text{Var}(\mu_x V_1 - \mu_y U_1).$$

**Proof** Let  $\mathbf{x} = (x, y) = (nh, mh)$ . By Theorem 1 we have

$$\rho_{X_\tau}(ih, jh) = G_{\sigma(m), \mu^*, N}^*(i) \Delta(jh) + o\left(\frac{1}{\sqrt{m}}\right),$$

where  $G_{\sigma(m), \mu^*, N}^*$  is defined by (47), has mean  $\mu^* = \hat{n} := \hat{x}/h$  and variance  $\sigma(m)^2 = \frac{\gamma^2 m}{|\mu_y|^3} := \tilde{\sigma}^2 m$ . By Lemma 1 statement ii, we have

$$G_{\sigma(m), \hat{n}, N}^*(i) = g_{\sigma(m), \hat{n}, N}^*(i) + O(m^{\frac{1}{2}} e^{-cm}),$$

for some constant  $c > 0$ , and hence

$$\rho_{X_\tau}(ih, jh) = g_{\sigma(m), \hat{n}, N}^*(i) \Delta(jh) + o\left(\frac{1}{\sqrt{m}}\right),$$

since the asymptotic decay of  $m^{\frac{1}{2}} e^{-cm}$  is strictly faster than that of  $m^{-\frac{1}{2}}$  for all  $c > 0$ . But  $h = y/m$  and  $y$  is a constant, so  $o(\frac{1}{\sqrt{m}}) = o(\sqrt{h})$ . Finally, since

$$\begin{aligned} g_{\sigma(m), \hat{n}, N}^*(i) &= \frac{e^{-\frac{d_N^{\text{circ}}(i, \hat{n})^2}{2\tilde{\sigma}^2 m}}}{\sum_{j=0}^{N-1} e^{-\frac{d_N^{\text{circ}}(j, \hat{n})^2}{2\tilde{\sigma}^2 m}}} \\ &= \frac{e^{-\frac{d^{\text{circ}}(ih, \hat{x})^2}{2\tilde{\sigma}^2 y h}}}{\sum_{j=0}^{N-1} e^{-\frac{d^{\text{circ}}(jh, \hat{x})^2}{2\tilde{\sigma}^2 y h}}} = g_{\sigma(y, h), \hat{x}, 1}^*(ih), \end{aligned}$$

we conclude

$$\rho_{X_\tau}(ih, jh) = g_{\sigma(y, h), \hat{x}, 1}^*(ih) \Delta(jh) + o(\sqrt{h}).$$

□

Now we move onto task three and derive an expression for  $\Delta(x_2)$  for the random walk  $\mathbf{X}_\tau$  under consideration:

**Lemma 4** Let  $H$  denote the law of  $V_{n_1}$ , where  $n_1 = \inf\{k : V_k < 0\}$ , and let  $v_y = E[V_{n_1}]$ . Let  $\Delta$  denote the asymptotic law of the overshoot  $Y_\tau$ . Then, for the direct method we have

$$\Delta(kh) = \frac{1}{\mu_y} \frac{\sum_{\mathbf{j} \in \bar{b}_r^-} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot \mathbf{e}_2 < k)}{W}.$$

For the semi-implicit method we have

$$\Delta(kh) = \frac{1}{v_y} \frac{\sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot \mathbf{e}_2 < k \cap \mathbf{j} \cdot \mathbf{e}_2 \neq 0)}{\sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot \mathbf{e}_2 \neq 0)},$$

where  $v_y = (1 - p_s)^{-1} \mu_y$  and  $p_s$  is the probability that the random walk in the implicit case does not move downwards in a given step:

$$p_s = \frac{1}{W} \sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot \mathbf{e}_2 = 0).$$

**Proof** First, note that since the increments  $\{V\}$  obey  $-r - 1 \leq V_i \leq 0$  for both the direct and semi-implicit methods, we have

$$\Delta(kh) = H((-\infty, k))/v_y = H([-r - 1, k))/v_y.$$

For the direct method, the ladder process above is particularly simple, since  $\mathbf{X}_\theta$  is guaranteed to move at least one pixel down in the  $y$  direction every iteration. In this case we simply have  $n_i = i$  for all  $i$ . Hence  $v_y = \mu_y$  and  $H$  is equal to the law of  $V_1$ . It follows that

$$\begin{aligned} \Delta(kh) &= \frac{1}{\mu_y} H([-r - 1, k)) \\ &= \frac{1}{\mu_y} \frac{\sum_{\ell=-r-1}^{k-1} \sum_{\mathbf{j} \in \bar{b}_r^-} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot \mathbf{e}_2 = \ell)}{W} \\ &= \frac{1}{\mu_y} \frac{\sum_{\mathbf{j} \in \bar{b}_r^-} \sum_{\ell=-r-1}^{k-1} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot \mathbf{e}_2 = \ell)}{W} \\ &= \frac{1}{\mu_y} \frac{\sum_{\mathbf{j} \in \bar{b}_r^-} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot \mathbf{e}_2 < k)}{W}. \end{aligned}$$

The semi-implicit case is more complex, since although  $X_\tau$  can never move in the positive  $y$  direction, it can “pause” at a constant value of  $y$  for an arbitrarily long duration before resuming its downward march. Let  $p_s$  denote the probability that a given increment of the walk does not move in the  $y$ -direction, that is  $p_s = P(V_i = 0)$ . Then we clearly have

$$p_s = \frac{1}{W} \sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot \mathbf{e}_2 = 0).$$

By Wald's Identity we have  $v_y = E[V_{n_1}] = \mu_y E[n_1]$  since  $n_1$  is a stopping time. We have

$$\begin{aligned} E[n_1] &= \sum_{n=1}^{\infty} p_s^{n-1} (1 - p_s) n \\ &= \frac{1 - p_s}{p_s} \sum_{n=1}^{\infty} n p_s^n \\ &= \frac{1 - p_s}{p_s} \cdot \frac{p_s}{(1 - p_s)^2} = \frac{1}{1 - p_s}, \end{aligned}$$

and hence

$$v_y = \frac{\mu_y}{1 - p_s}.$$

The law of  $V_{n_1}$  (and hence  $H$ ) is given by:

$$p_{V_{n_1}}(\mathbf{j}) = \frac{\tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot e_2 \neq 0)}{\sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot e_2 \neq 0)}.$$

It follows that

$$\begin{aligned} \Delta(kh) &= \frac{1}{v_y} H([-r - 1, k)) \\ &= \frac{1}{\mu_y} \frac{\sum_{\ell=-r-1}^{k-1} \sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot e_2 = \ell)}{\sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot e_2 \neq 0)} \\ &= \frac{1}{v_y} \frac{\sum_{\mathbf{j} \in \bar{b}_r^0} \sum_{\ell=-r-1}^{k-1} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot e_2 = \ell)}{\sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot e_2 \neq 0)} \\ &= \frac{1}{v_y} \frac{\sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot e_2 < k)}{\sum_{\mathbf{j} \in \bar{b}_r^0} \tilde{w}_r(\mathbf{0}, \mathbf{j}) 1(\mathbf{j} \cdot e_2 \neq 0)}. \end{aligned}$$

□

Our final task before proving our main result is to bound the probability of  $\mathbf{X}_\tau$  deviating significantly from  $\Pi_{\theta_r^*}(\mathbf{x})$ . The following Lemma accomplishes this.

**Lemma 5** *Let  $\mathbf{x} \in D_h$  and  $\hat{\mathbf{x}} := (\hat{x}, 0) := \Pi_{\theta_r^*}(\mathbf{x})$ . Let  $d_1^{\text{circ}}$  denote the circular distance (46). Then*

$$\begin{aligned} P(d_1^{\text{circ}}(X_\tau, \hat{x}) > \delta) &\leq 2 \exp \left( - \frac{(\delta - rh)^2 \sin^2 \theta_r^*}{4 \frac{r}{|\mu_y|} rh} \right) \\ &\quad + \exp \left( - \frac{1}{4 \frac{r}{|\mu_y|} rh} \right). \end{aligned}$$

**Proof** Our random walk on the periodic domain  $[0, 1) \times (-\delta, 1]$  is equivalent to the same walk on the infinite domain  $\mathbb{R} \times (-\delta, 1]$ , modulo the equivalence relation  $(x, y) \sim (x + k, y)$  for all  $k \in \mathbb{Z}$ . Let us denote (within this lemma only) our periodic walk by  $\mathbf{X}_\tau^*$  while the walk on  $\mathbb{R} \times (-\delta, 1]$  is denoted by  $\mathbf{X}_\tau$ . Then we have

$$\begin{aligned} \{d_1^{\text{circ}}(\mathbf{X}_\tau^*, \hat{x}) > \delta\} &= \bigcap_{k \in \mathbb{Z}} \{|X_\tau - (\hat{x} + k)| > \delta\} \\ &\subseteq \{|X_\tau - \hat{x}| > \delta\}. \end{aligned}$$

We therefore have

$$P(d_1^{\text{circ}}(\mathbf{X}_\tau^*, \hat{x}) > \delta) \leq P(|X_\tau - \hat{x}| > \delta),$$

and hence it suffices to prove

$$\begin{aligned} P(|X_\tau - \hat{x}| > \delta) &\leq 2 \exp\left(-\frac{(\delta - rh)^2 \sin^2 \theta_r^*}{4 \frac{r}{|\mu_y|} rh}\right) \\ &\quad + \exp\left(-\frac{1}{4 \frac{r}{|\mu_y|} rh}\right). \end{aligned}$$

*Step 1.*  $P\left(\tau > \lceil \frac{2}{|\mu_y|h} \rceil\right) \leq \exp\left(-\frac{1}{4 \frac{r}{|\mu_y|} rh}\right)$ . We define

$$M_k := Y_{k \wedge \tau} - (\tau \wedge k) \mu_y h - mh$$

which the reader may verify is a zero mean martingale with bounded increments

$$|M_{k+1} - M_k| \leq rh.$$

Next we note that for any  $k$  the following events are equal:

$$\begin{aligned} \{\tau \geq k\} &= \{Y_{\tau \wedge k} \geq 0\} = \{M_k \geq -(k \wedge \tau) \mu_y h - mh\} \\ &= \{M_k \geq -k \mu_y h - mh\}. \end{aligned}$$

Therefore

$$\begin{aligned} \left\{\tau \geq \left\lceil \frac{2}{|\mu_y|h} \right\rceil\right\} &= \left\{M_{\lceil \frac{2}{|\mu_y|h} \rceil} \geq -\left\lceil \frac{2}{|\mu_y|h} \right\rceil \mu_y h - mh\right\} \\ &\subseteq \left\{M_{\lceil \frac{2}{|\mu_y|h} \rceil} \geq 1\right\} \end{aligned}$$

where we have used the inequality

$$-\left\lceil \frac{2}{|\mu_y|h} \right\rceil \mu_y h - mh = \left\lceil \frac{2}{|\mu_y|h} \right\rceil |\mu_y|h - mh \geq 2 - mh \geq 1.$$



Noting that  $M_0 = 0$  we apply Azuma's inequality to find

$$\begin{aligned} P\left(\tau > \left\lceil \frac{2}{|\mu_y|h} \right\rceil\right) &\leq P\left(M_{\left\lceil \frac{2}{|\mu_y|h} \right\rceil} \geq 1\right) \\ &\leq \exp\left(-\frac{1}{2\left\lceil \frac{2}{|\mu_y|h} \right\rceil r^2 h^2}\right) \\ &\leq \exp\left(-\frac{1}{4\frac{r}{|\mu_y|}rh}\right). \end{aligned}$$

*Step 2.* Let  $d(\mathbf{X}_\tau, L_{\mathbf{x}, \hat{\mathbf{x}}})$  denote the orthogonal distance from  $\mathbf{X}_\tau$  to the line passing through  $\mathbf{x}$  and  $\hat{\mathbf{x}}$ . This time we define

$$M_k := (\mathbf{X}_{\tau \wedge k} - \mathbf{x}) \cdot (-\sin \theta_r^*, \cos \theta_r^*).$$

Once again,  $M_k$  is a zero-mean martingale with bounded increments  $|M_{k+1} - M_k| \leq rh$  obeying  $M_0 = 0$ . Therefore, applying Azuma's inequality again gives, for any  $k \in \mathbb{N}$ ,

$$P(|M_k| > \delta) \leq 2 \exp\left(-\frac{\delta^2}{2kr^2h^2}\right).$$

Moreover, if  $\tau \leq k$ , then we have the equality

$$|M_k| = d(\mathbf{X}_\tau, L_{\mathbf{x}, \hat{\mathbf{x}}})$$

For any integer  $k \in \mathbb{N}$  we clearly have

$$\begin{aligned} P(d(\mathbf{X}_\tau, L_{\mathbf{x}, \hat{\mathbf{x}}}) > \delta) &\leq P(\{d(\mathbf{X}_\tau, L_{\mathbf{x}, \hat{\mathbf{x}}}) > \delta\} \cap \{\tau \leq k\}) \\ &\quad + P(\tau > k) \\ &\leq P(|M_k| > \delta) + P(\tau > k). \end{aligned}$$

Taking  $k = \left\lceil \frac{2}{|\mu_y|h} \right\rceil$ , substituting the above bound as well as the one from Step 1 gives

$$\begin{aligned} P(d(\mathbf{X}_\tau, L_{\mathbf{x}, \hat{\mathbf{x}}}) > \delta) &\leq 2 \exp\left(-\frac{\delta^2}{4\frac{r}{|\mu_y|}rh}\right) \\ &\quad + \exp\left(-\frac{1}{4\frac{r}{|\mu_y|}rh}\right). \end{aligned}$$

Finally, by the triangle inequality and simple trigonometry we have

$$|X_\tau - \hat{x}| \leq \frac{d(\mathbf{X}_\tau, L_{\mathbf{x}, \hat{\mathbf{x}}})}{\sin \theta_r^*} + rh.$$

Hence

$$P(|X_\tau - \hat{x}| > \delta) \leq P(d(\mathbf{X}_\tau, L_{\mathbf{x}, \hat{\mathbf{x}}}) > (\delta - rh) \sin \theta_r^*),$$

from which the claimed result follows trivially.  $\square$

Having accomplished all of our tasks, we are now ready to prove our main result.

**Theorem 2** *Let the inpainting domain  $D$  and undamaged area  $\mathcal{U}$ , as well as their discrete counterparts  $D_h, \mathcal{U}_h$  be as described in Sect. 6.1. Let  $u_0 : \mathcal{U} \rightarrow \mathbb{R}^d$  denote as usual the undamaged portion of the image, and assume  $u_0$  is bounded in norm, that is, there is an  $M > 0$  such that  $\|u_0\| \leq M$  on  $\mathcal{U}$ . Suppose we inpaint  $D_h$  using Algorithm 1 or its semi-implicit extension, and denote the result by  $u_h : D_h \rightarrow \mathbb{R}^d$ . Assume the assumptions of Sect. 6.1 hold and let  $a_r^*$  denote the stencil of our inpainting method. Let  $\mathbf{Z}_i = (U_i, V_i)$  taking values in  $\text{Supp}(a_r^*)$  with mean  $(\mu_x, \mu_y)$  denote the increments of the random walk described above, with probability density given by (35). Assume that  $a_r^*$  is not confined to any sublattice of  $\mathbb{Z}^2$  in the sense of Definition 3, or equivalently that the characteristic function  $\varphi_{\mathbf{Z}_i}$  of  $\mathbf{Z}_i$ , that is*

$$\varphi_{\mathbf{Z}_i}(\mathbf{u}) = \sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \frac{w_r(\mathbf{0}, \mathbf{y}) e^{\sqrt{-1} \mathbf{y} \cdot \mathbf{u}}}{W},$$

*obeys  $|\varphi_{\mathbf{Z}_i}(\mathbf{u})| < 1$  unless both components of  $\mathbf{u}$  are integer multiples of  $2\pi$ . Let  $\Pi_{\partial_r^*} : D \rightarrow \partial D$  denote the transport operator defined in (32) (recall that the transport direction  $\mathbf{g}_r^*$  obeys  $\mathbf{g}_r^* = (\mu_x, \mu_y)$ ), and let  $\mathbf{x} = (x, y) = (nh, mh) \in D_h$ , and assume  $y$  is fixed. Let*

$$\tilde{g}_{\sigma(y,h)}(ih, jh) = g_{\sigma(y,h),1}^*(ih) \bar{\Delta}(jh)$$

*where  $g_{\sigma(y,h),1}^*$  is the one-dimensional discrete Gaussian*

$$g_{\sigma(y,h)}(ih) = \frac{e^{-\frac{d_1^{\text{circ}}(ih,0)^2}{2\sigma(y,h)^2}}}{\sum_{k=0}^{N-1} e^{-\frac{d_1^{\text{circ}}(kh,0)^2}{2\sigma(y,h)^2}}}$$

*with variance*

$$\sigma(y,h)^2 = \frac{\gamma^2 y h}{|\mu_y|^3} \quad \text{where } \gamma^2 = \text{Var}(\mu_x V_1 - \mu_y U_1)$$

*and where  $\bar{\Delta}$  denotes the even extension of  $\Delta$  defined by*

$$\bar{\Delta}(jh) = \Delta(|j|h)$$

where  $\Delta(jh)$  is given by Lemma 4. Then

$$u_h(x, y) = (u_0 * \tilde{g}_{\sigma(y,h)})(\Pi_{\theta_r^*}(x, y)) + o(1) \quad (51)$$

**Proof** Our goal is to show that for any  $\epsilon > 0$ , taking  $h$  sufficiently small gives

$$\|u_h(x, y) - (u_0 * \tilde{g}_{\sigma(y,h)})(\Pi_{\theta_r^*}(x, y))\| < \epsilon.$$

Let us define  $\hat{\mathbf{x}} := (\hat{x}, 0) := \Pi_{\theta_r^*}(x, y)$ . Then this is equivalent to proving

$$\left\| \sum_{\mathbf{x} \in \mathcal{U}_h} u_0(\mathbf{x}) (\rho_{\mathbf{X}_\tau}(\mathbf{x}) - \tilde{g}_{\sigma(y,h)}(\hat{\mathbf{x}} - \mathbf{x})) \right\| < \epsilon,$$

where the  $x$ -coordinate of  $\hat{\mathbf{x}} - \mathbf{x}$  is understood to be taken modulo 1 due to our periodic boundary conditions. Since  $\|u_0\| \leq M$  it suffices to prove

$$\begin{aligned} S &\equiv \sum_{(ih,jh) \in \mathcal{U}_h} \left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right| \\ &< \frac{\epsilon}{M}. \end{aligned}$$

We will prove shortly that the per-term error

$$\left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right|$$

is of size  $o(1/\sqrt{N})$ . However, this condition alone is not enough to prove the desired result, as  $S$  contains  $O(N)$  terms. To get around this, we introduce the sets  $I = \{0, 1, \dots, N-1\}$  and

$$I_k = \{i \bmod N\}_{i=\lceil \frac{\hat{x}}{h} - k\sqrt{m} \rceil}^{\lfloor \frac{\hat{x}}{h} + k\sqrt{m} \rfloor},$$

where  $k \in \mathbb{N}$  is a constant to be determined. Note that  $I_k$  may equivalently be characterized as

$$I_k = \left\{ i \in I : d_N^{\text{circ}}\left(i, \frac{\hat{x}}{h}\right) \leq k\sqrt{m} \right\},$$

which in turn means that

$$I \setminus I_k = \left\{ i \in I : d_N^{\text{circ}}\left(i, \frac{\hat{x}}{h}\right) > k\sqrt{m} \right\}, \quad (52)$$

Which will be critical later.

We break the sum into two pieces  $S = S_1 + S_2$  defined as

$$S_1 \equiv \sum_{j=-r-2}^0 \sum_{i \in I_k} \left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right|$$

$$S_2 \equiv \sum_{j=-r-2}^0 \sum_{i \in I \setminus I_k} \left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right|.$$

Since  $m = yN$  and  $y$  is fixed,  $m$  is proportional to  $N$ , and hence  $S_1$  contains  $O(\sqrt{m}) = O(\sqrt{N})$  terms concentrated around  $\hat{x}$ . We will control the size of  $S_1$  by using the  $o(\sqrt{h}) = o(1/\sqrt{N})$  error estimate from Lemma 3. The second sum  $S_2$  will contain all remaining terms in  $S$ . We will bound  $S_2$  by making separate tail estimates for  $\rho_{\mathbf{X}_\tau}$  and  $g_{\sigma(y,h),1}^*(\hat{x} - ih)$ , both based on (52).  $\square$

**Claim 1** For  $k \geq \max\left(\frac{2r}{\sqrt{y}}, 4\right)$  there exist constants  $C_1, C_2, C_3, C_4 > 0$  (dependent on  $r$  and  $y$ ) such that

$$S_2 < C_1 e^{-C_2 k^2} + C_3 N^{\frac{3}{2}} e^{-C_4 N}.$$

We defer the proof of Claim 1 until the end. For now, note that it gives us the estimate

$$S \leq S_1 + C_1 e^{-C_2 k^2} + C_3 N^{\frac{3}{2}} e^{-C_4 N}.$$

Fixing  $k$  so that  $C_1 e^{-C_2 k^2} < \frac{\epsilon}{4M}$  and  $k \geq \frac{2r}{\sqrt{y}}$  are both true, and taking  $N$  large enough that  $C_3 N^{\frac{3}{2}} e^{-C_4 N} < \frac{\epsilon}{4M}$ , it remains to prove that  $S_1 < \frac{\epsilon}{2M}$ . Defining  $C := (r+3)3k\sqrt{y}$  we have

$$S_1 \equiv \sum_{j=-r-2}^0 \sum_{i \in I_k} \left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right|$$

$$\leq C\sqrt{N} \sup_{i,j} \left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right|.$$

By Lemma 3, we have

$$\sqrt{N} \sup_{i,j} \left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right|$$

$$\rightarrow 0 \quad \text{as } N \rightarrow \infty.$$

Taking making  $N$  large enough (equivalently,  $h$  small enough) that

$$\sqrt{N} \sup_{i,j} \left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right| < \frac{1}{C} \frac{\epsilon}{2M},$$

we are done.

**Proof of Claim 1** We break  $S_2$  into two pieces:

$$\begin{aligned} S_2 &\equiv \sum_{j=-r-2}^0 \sum_{i \in I \setminus I_k} \left| \rho_{\mathbf{X}_\tau}(ih, jh) - \Delta(jh) g_{\sigma(y,h),1}^*(\hat{x} - ih) \right| \\ &\leq \sum_{j=-r-2}^0 \sum_{i \in I \setminus I_k} \rho_{\mathbf{X}_\tau}(ih, jh) \\ &\quad + \sum_{j=-r-2}^0 \Delta(jh) \sum_{i \in I \setminus I_k} g_{\sigma(y,h),1}^*(\hat{x} - ih) \end{aligned}$$

But by (52), we have

$$\begin{aligned} S_2 &\leq P(d_1^{\text{circ}}(X_\tau, \hat{x}) > k\sqrt{y}\sqrt{N}h) \\ &\quad + \sum_{i \in I \setminus I_k} g_{\sigma(y,h),1}^*(\hat{x} - ih). \end{aligned}$$

The first of these, we already know how to bound. By Lemma 5, we have

$$\begin{aligned} &P(d_1^{\text{circ}}(X_\tau, \hat{x}) > k\sqrt{y}\sqrt{N}h) \\ &\leq 2 \exp \left( - \frac{(k\sqrt{y}\sqrt{N}h - rh)^2 \sin^2 \theta_r^*}{4r^2/|\mu_y|h} \right) \\ &\quad + \exp \left( - \frac{1}{4 \frac{r}{|\mu_y|} rh} \right) \end{aligned}$$

Since  $k \geq \frac{2r}{\sqrt{y}}$  we have

$$\frac{(k\sqrt{y}\sqrt{N}h - rh)^2}{h} = (k\sqrt{y} - r\sqrt{h})^2 \geq (k\sqrt{y} - r)^2 \geq 0.5yk^2,$$

it follows that

$$\begin{aligned} &P(d_1^{\text{circ}}(X_\tau, \hat{x}) > k\sqrt{y}\sqrt{N}h) \\ &\leq 2 \exp \left( -y \left\{ \frac{\sin^2 \theta_r^*}{16r^2/|\mu_y|} \right\} k^2 \right) \\ &\quad + \exp \left( - \frac{1}{4 \frac{r}{|\mu_y|} rh} \right). \end{aligned} \tag{53}$$

For the second, we find it convenient to rewrite the sum in terms of  $G_{\sigma(m),N}^*(i - \hat{x}/h)$ . To that end, we define  $\tilde{\sigma}^2 = \sigma^2(y, h)/(yh) = \frac{\gamma^2}{|\mu_y|^3}$  where  $\gamma^2 > 0$  is a constant given by (45). Then

$$\begin{aligned}
\sum_{i \in I \setminus I_k} g_{\sigma(y,h),1}^*(ih - \hat{x}) &= \sum_{i \in I \setminus I_k} g_{\sigma(m),N}(i - \hat{x}/h) \\
&\leq \sum_{i \in I \setminus I_k} G_{\sigma(m),N}^*(i - \hat{x}/h) \\
&\quad + \frac{A}{y} m^{\frac{3}{2}} e^{-bm},
\end{aligned} \tag{54}$$

where we have used Lemma 1 together with  $|I \setminus I_k| \leq N = m/y$  in the rightmost inequality. Next, define

$$\mathcal{E} = \left\{ d_N^{\text{circ}} \left( i, \frac{\hat{x}}{h} \right) : i \in I \setminus I_k \right\}.$$

Since  $I \setminus I_k \subseteq \mathbb{Z}$ , every member of  $\mathcal{E}$  must be of the form

$$\xi + n \quad \text{where } \xi = \frac{\hat{x}}{h} - \left\lfloor \frac{\hat{x}}{h} \right\rfloor \in [0, 1)$$

for some  $n \in \mathbb{Z}$ . But by (52), we must have  $n \geq \lfloor k\sqrt{m} \rfloor$ . Finally, since the map  $i \in I \setminus I_k \rightarrow d_N^{\text{circ}}(i, \frac{\hat{x}}{h})$  is at most two to one, we have

$$\begin{aligned}
\sum_{i \in I \setminus I_k} G_{\sigma(m),N}^*(i - \hat{x}/h) &= \sum_{i \in I \setminus I_k} G_{\sigma(m)}(d_N^{\text{circ}}(i, \hat{x}/h)) \\
&\leq 2 \sum_{x \in \mathcal{E}} G_{\sigma(m)}(x) \\
&\leq 2 \sum_{n=\lfloor k\sqrt{m} \rfloor}^{\infty} G_{\sigma(m)}(\xi + n)
\end{aligned}$$

The following routine tail estimate gives us a bound on our sum.

$$\begin{aligned}
\sum_{i \in I \setminus I_k} G_{\sigma(m),N}^*(i - \hat{x}/h) &\leq 2 \sum_{n=\lfloor k\sqrt{m} \rfloor}^{\infty} G_{\sigma(m)}(\xi + n) \\
&= 2 \sum_{n=\lfloor k\sqrt{m} \rfloor}^{\infty} \frac{e^{-\frac{(\xi+n)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} \\
&\leq 2 \sum_{n=\lfloor k\sqrt{m} \rfloor}^{\infty} \frac{e^{-\frac{n^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} \\
&\leq 2 \int_{\lfloor k\sqrt{m} \rfloor - 1}^{\infty} \frac{e^{-\frac{x^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} dx.
\end{aligned}$$

Since  $k > 4$ , we can find a  $c \in (0, 1)$  independent of  $1 \leq m \leq N$  and  $k$  (for example  $c = 0.5$  will work) such that  $ck\sqrt{m} \leq \lfloor k\sqrt{m} \rfloor - 1$ . At the same time, we have  $\sigma(m)^2 = \tilde{\sigma}^2 m$  where  $\tilde{\sigma}^2 = \gamma^2 / |\mu_y|^3$  with  $\gamma^2 = \text{Var}(\mu_x V_1 - \mu_y U_1)$ . Therefore

$$\begin{aligned} \sum_{i \in I \setminus I_k} G_{\sigma(m), N}^*(i - \hat{x}/h) &\leq 2 \int_{ck\sqrt{m}}^{\infty} \frac{e^{-\frac{x^2}{2\tilde{\sigma}^2 m}}}{\sqrt{2\pi\tilde{\sigma}\sqrt{m}}} dx \\ &\leq 2 \int_{ck\sqrt{m}}^{\infty} \frac{x}{ck\sqrt{m}} \frac{e^{-\frac{x^2}{2\tilde{\sigma}^2 m}}}{\sqrt{2\pi\tilde{\sigma}\sqrt{m}}} dx \\ &= \sqrt{\frac{2}{\pi}} \frac{\tilde{\sigma}}{ck} e^{-\frac{c^2 k^2}{2\tilde{\sigma}^2}} \\ &\leq \sqrt{\frac{2}{\pi}} \frac{\tilde{\sigma}}{c} e^{-\frac{c^2 k^2}{2\tilde{\sigma}^2}} \end{aligned}$$

Putting this together with (54) and (53) gives

$$\begin{aligned} S_2 &\leq 2 \exp\left(-y \left\{ \frac{\sin^2 \theta_r^*}{4r^2/|\mu_y|} \right\} k^2\right) + \exp\left(-\frac{1}{4\frac{r}{|\mu_y|}rh}\right) \\ &\quad + \sqrt{\frac{2}{\pi}} \frac{\tilde{\sigma}}{c} e^{-\frac{c^2 k^2}{2\tilde{\sigma}^2}} + A m^{\frac{3}{2}} e^{-bm} \\ &\leq \left(2 + \sqrt{\frac{2}{\pi}} \frac{\tilde{\sigma}}{c}\right) e^{-\alpha k^2} + (1+A)y^{\frac{3}{2}} N^{\frac{3}{2}} e^{-\beta y N} \end{aligned}$$

where  $\alpha \equiv \min(\frac{\sin^2 \theta_r^*}{4r^2/|\mu_y|}, \frac{c^2}{2\tilde{\sigma}^2})$ ,  $\beta \equiv \min(\frac{1}{4r^2/|\mu_y|}, b)$ . Taking  $C_1 = \left(2 + \sqrt{\frac{2}{\pi}} \frac{\tilde{\sigma}}{c}\right)$ ,  $C_2 = \alpha$ ,  $C_3 = (1+A)y^{\frac{3}{2}}$ ,  $C_4 = \beta y$  completes the proof of the claim.  $\square$

## 6.6 Asymptotic Limit Version 2

An important special case of the general framework considered in the previous two sections occurs when only the  $x$ -component of  $\mathbf{X}_\tau$  is random. In particular, let us assume that the increments  $\Delta \mathbf{X}_i = h \mathbf{Z}_i = h \cdot (U, V)$  obey  $U \in \{-(r+2), -(r+1), \dots, r+1, r+2\}$  but  $V \equiv -1$ . Then the assumptions of Theorem 1 are *never* satisfied, as

$$\varphi_{\mathbf{Z}_i}(u_1, u_2) = e^{-\sqrt{-1}u_2} \sum_{k=-(r+2)}^{(r+2)} P(U=k) e^{\sqrt{-1}ku_1}$$

obeys  $|\varphi_{\mathbf{Z}_i}| = 1$  for  $u_1$  an integer multiple of  $2\pi$  and any  $u_2 \in \mathbb{R}$ . Since we cannot use Theorem 1, a different approach is required in this case.

Here we use the following local central limit theorem for  $U$ :

**Theorem 3** *Let  $X_1, X_2, \dots$  be i.i.d. copies of an integer-valued random variable  $X$  of mean  $\mu$  and variance  $\sigma^2$ . Suppose furthermore that there is no infinite subprogression  $a + q\mathbb{Z}$  of  $\mathbb{Z}$  with  $q > 1$  for which  $X$  takes values almost surely in  $a + q\mathbb{Z}$ . Then one has*

$$\mathbf{P}(S_n = m) = \frac{1}{\sqrt{2\pi n\sigma}} e^{-(m-n\mu)^2/2n\sigma^2} + o(1/n^{1/2})$$

for all  $n \geq 1$  and all integers  $m$ , where the error term  $o(1/n^{1/2})$  is uniform in  $m$ .

**Proof** This theorem may be found on Terrance Tao's blog [37].  $\square$

The requirement in Theorem 3 “that there is no infinite subprogression  $a + q\mathbb{Z}$  of  $\mathbb{Z}$  with  $q > 1$  for which  $X$  takes values almost surely in  $a + q\mathbb{Z}$ ” is equivalent to the following one dimensional version of Definition 3:

**Definition 4** We say that an integer valued random variable  $X$  is “confined to a sublattice of  $\mathbb{Z}$ ” if there exists an  $n \in \mathbb{Z}$  and  $k \in \mathbb{Z}$  with  $|k| > 1$ , such that the density of  $X$  is zero outside of  $n + k\mathbb{Z}$ . We say that  $X$  is not confined to any sublattice of  $\mathbb{Z}$  if no such  $n$  and  $k$  exist.

Moreover, by an argument almost identical to that of Lemma 2, this is equivalent to the characteristic function  $\varphi_X(u)$  defined by

$$\varphi_X(u) = \sum_{j \in \mathbb{Z}} P(X = j) e^{\sqrt{-1}ju}$$

obeying  $|\varphi_X(u)| < 1$  unless  $u$  is an integer multiple of  $2\pi$ .

Following a sequence of steps almost identical to the previous section, we arrive at a version of the asymptotic limit applicable to this case.

**Theorem 4** *Let the inpainting domain  $D$  and undamaged area  $\mathcal{U}$ , as well as their discrete counterparts  $D_h, \mathcal{U}_h$  be as described in Sect. 6.1. Let  $u_0 : \mathcal{U} \rightarrow \mathbb{R}^d$  denote as usual the undamaged portion of the image, and assume  $u_0$  is bounded in norm, that is, there is an  $M > 0$  such that  $\|u_0\| \leq M$  on  $\mathcal{U}$ . Suppose we inpaint  $D_h$  using Algorithm 1 or its semi-implicit extension, and denote the result by  $u_h : D_h \rightarrow \mathbb{R}^d$ . Assume the assumptions of Sect. 6.1 hold and let  $a_r^*$  denote the stencil of our inpainting method. Assume that the stencil weights  $\tilde{w}_j$  assign zero mass outside of the line  $\mathbb{Z} \times \{-1\}$ , and assume that the  $x$ -coordinate  $U$  of  $\mathbf{Z}_i = (U, V)$  is not confined to any sublattice of  $\mathbb{Z}$  in the sense Definition 4, or equivalently that the characteristic function  $\varphi_U$  of  $U$  defined by*

$$\varphi_U(u) = \sum_{j=-(r+2)}^{r+2} P(U = j) e^{\sqrt{-1}ju}$$



obeys  $\varphi_U(u) < 1$  unless  $u$  is an integer multiple of  $2\pi$ . Let  $\Pi_{\theta_r^*} : D \rightarrow \partial D$  denote the transport operator defined in (32) (recall that the transport direction  $\mathbf{g}_r^*$  obeys  $\mathbf{g}_r^* = (\mu_x, \mu_y)$ ), and let  $\mathbf{x} = (x, y) \in D_h$ . Let  $g_{\sigma(y,h)}$  denote the one-dimensional discrete Gaussian

$$g_{\sigma(y,h)}(ih) = \frac{e^{-\frac{d_1^{\text{circ}}(ih,0)^2}{2\sigma(y,h)^2}}}{\sum_{k=0}^{N-1} e^{-\frac{d_1^{\text{circ}}(kh,0)^2}{2\sigma(y,h)^2}}}$$

with mean 0 and variance

$$\sigma(y, h) = \sigma^2 y h$$

where  $\sigma^2$  denote the variance of  $U$ . Then

$$u_h(x, y) = (u_0 * g_{\sigma(y,h)})(\Pi_{\theta_r^*}(x, y)) + o(1) \quad (55)$$

**Proof** The proof is nearly identical to that of Theorem 2 and is left as an exercise to the reader.  $\square$

## 6.7 Derivation of Fixed-Ratio Continuum Limit

In this section, we prove pointwise convergence of the discrete solution  $u_h(\mathbf{x})$  to the fixed ratio continuum limit  $u(\mathbf{x})$ , for those points  $\mathbf{x} \in D$  for which  $\Pi_{\theta_r^*}(\mathbf{x}) \in \partial D$  is a continuity point of  $u_0$ . Theorem 2 tells us that

$$u_h(\mathbf{x}) = \tilde{g}_{\sigma(y,h)} * u_0(\Pi_{\theta_r^*}(\mathbf{x})) + o(1),$$

where  $\tilde{g}_{\sigma(y,h)}$  is the blur kernel from Lemma 3. Since  $u(\mathbf{x}) = u_0(\Pi_{\theta_r^*}(\mathbf{x}))$ , our job amounts to proving that

$$\tilde{g}_{\sigma(h)} * u_0(\mathbf{x}) \rightarrow u_0(\mathbf{x}) \quad \text{as } h \rightarrow 0$$

when  $\mathbf{x}$  is a continuity point of  $u_0$ .

Before tackling this problem, it is worth digressing briefly to consider what would happen had we considered the original high resolution vanishing viscosity limit of März and Bornemann, in which first  $h \rightarrow 0$  and then  $\epsilon \rightarrow 0$ . Writing  $u_{\epsilon,h}$  in place of  $u_h$  to make the dependence on  $\epsilon$  explicit, after  $h \rightarrow 0$ , we are left with

$$u_\epsilon = \lim_{h \rightarrow 0} u_{\epsilon,h} = G_{\sigma(y,\epsilon)} * u_0$$

where  $G_{\sigma(y,\epsilon)}$  is an integral convolution operator. For  $u_0 \in BV(\mathcal{U})$  the limit

$$\lim_{\epsilon \rightarrow 0} u_\epsilon(\mathbf{x})$$

can be almost characterized completely by well known results for functions of bounded variation. For completeness, we quote these results and the relevant definitions here:

**Definition 5** (*Approximate limit*) we say that  $u \in BV(\Omega)$  has an approximate limit at  $\mathbf{x} \in \Omega$  if there is a  $z \in \mathbb{R}$  such that

$$\lim_{\rho \downarrow 0} \frac{1}{|B_\rho(\mathbf{x})|} \int_{B_\rho(\mathbf{x})} |u(y) - z| dy = 0.$$

The set  $S_u$  of points where this property does not hold is called the approximate discontinuity set. For any  $\mathbf{x} \in \Omega \setminus S_u$ , we define  $\tilde{u}(\mathbf{x}) = z$ .

**Definition 6** (*Approximate jump points*) Given  $\mathbf{x} \in \Omega \subseteq \mathbb{R}^n$ ,  $\mathbf{v} \in S^{n-1}$ , and  $\rho > 0$ , define:

$$\begin{aligned} B_\rho^+(\mathbf{x}, \mathbf{v}) &:= \{\mathbf{y} \in B_\rho(\mathbf{x}) : (\mathbf{y} - \mathbf{x}) \cdot \mathbf{v} > 0\} \\ B_\rho^-(\mathbf{x}, \mathbf{v}) &:= \{\mathbf{y} \in B_\rho(\mathbf{x}) : (\mathbf{y} - \mathbf{x}) \cdot \mathbf{v} < 0\}. \end{aligned}$$

Let  $u \in BV(\Omega) \subseteq \mathbb{R}^n$  and  $\mathbf{x} \in \Omega$ . We say that  $\mathbf{x}$  is an approximate jump point of  $u$  if there exist  $a, b \in \mathbb{R}$  and  $\mathbf{v} \in \mathbb{R}^{n-1}$  such that  $a \neq b$  and

$$\begin{aligned} \lim_{\rho \downarrow 0} \frac{1}{|B_\rho^+(\mathbf{x})|} \int_{B_\rho^+(\mathbf{x})} |u(y) - a| dy &= 0. \\ \lim_{\rho \downarrow 0} \frac{1}{|B_\rho^-(\mathbf{x})|} \int_{B_\rho^-(\mathbf{x})} |u(y) - b| dy &= 0. \end{aligned}$$

The set of  $\mathbf{x} \in \Omega$  where this property holds is called the approximate jump set  $J_u$  of  $u$ . For any  $\mathbf{x} \in J_u$ , we define  $u^+(\mathbf{x}) = a$  and  $u^-(\mathbf{x}) = b$ .

**Theorem 5** If  $\rho_\epsilon$  is a family of mollifiers and if  $\mathbf{x} \in \Omega \setminus S_u$ , then the functions  $u * \rho_\epsilon(\mathbf{x})$  converge to  $\tilde{u}(\mathbf{x})$  as  $\epsilon \downarrow 0$ . Moreover, if  $\mathbf{x} \in J_u$ , then

$$\lim_{\epsilon \downarrow 0} u * \rho_\epsilon(\mathbf{x}) \rightarrow \frac{u^+(\mathbf{x}) + u^-(\mathbf{x})}{2}.$$

**Proof** See Proposition 3.64 and Corollary 3.80 on pages 160 and 175 respectively of [1].  $\square$

März and Bornemann's *high-resolution vanishing viscosity* limit is also the solution to a transport equation, differing from our fixed-ratio limit only in that the transport operator, which we denote here by  $\Pi_M$ , is different (see [7,31] for details). If  $u_0 \in SBV(\mathcal{U})$ , the approximate discontinuity set  $S_{u_0}$  equals the approximate jump set  $J_{u_0}$  and in this case we have

$$\lim_{\epsilon \rightarrow 0} u_\epsilon(\mathbf{x}) = u_M(\mathbf{x})$$

where

$$u_M = u_0(\Pi_M(\mathbf{x})) \text{ if } \mathbf{x} \in \mathcal{U} \setminus J_{u_0}$$

and

$$u_M = \alpha u_0^+(\Pi_M(\mathbf{x})) + (1 - \alpha) u_0^-(\Pi_M(\mathbf{x})) \text{ if } \mathbf{x} \in J_{u_0}$$

for some  $\alpha \in [0, 1]$ . This is a result that März and Bornemann appear to be unaware of, as their approach is very different from ours.

In our case, however,  $\tilde{g}_{\sigma(y,h)}$  is *discrete* and  $BV(\mathcal{U})$  (or  $SBV(\mathcal{U})$ ), where functions are only defined up to a set of measure zero, does not give us the right setup to prove our result. Moreover, even for piecewise continuous boundary data with simple jump type discontinuities, the behavior of  $u_0(\mathbf{x})$  when  $\Pi_{\theta_r^*}(\mathbf{x})$  is not a continuity point of  $u_0$  is in general rather complex. Therefore, we content ourselves with proving convergence in the case where  $\Pi_{\theta_r^*}(\mathbf{x})$  is a continuity point of  $u_0$ .

**Theorem 6** Assume  $u_0$  obeys the same conditions as in Theorem 2. Then for every  $\mathbf{x} \in D$  such that  $\Pi_{\theta_r^*}(\mathbf{x})$  is a continuity point of  $u_0$  we have

$$u_h(\mathbf{x}) \rightarrow u(\mathbf{x}) = u_0(\Pi_{\theta_r^*}(\mathbf{x})) \quad \text{as } h \rightarrow 0.$$

where  $\Pi_{\theta_r^*}(\mathbf{x})$  is given by

$$\Pi_{\theta_r^*}(x, y) = (x - \cot(\theta_r^*)y \mod 1, 0),$$

and

$$\theta_r^* = \theta(\mathbf{g}_r^*) \in (0, \pi)$$

is the counterclockwise angle between the x-axis and the line  $L_{\mathbf{g}_r^*} := \{\lambda \mathbf{g}_r^* : \lambda \in \mathbb{R}\}$ , and  $\mathbf{g}_r^*$  is the center of mass of the stencil  $a_r^*$  which can be computed in two equivalent ways (Definition 1)

$$\mathbf{g}_r^* = \frac{\sum_{\mathbf{y} \in a_r^*} w_r(0, \mathbf{y}) \mathbf{y}}{\sum_{\mathbf{y} \in a_r^*} w_r(0, \mathbf{y})} = \frac{\sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \tilde{w}_r(0, \mathbf{y}) \mathbf{y}}{\sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \tilde{w}_r(0, \mathbf{y})}.$$

**Proof** Assume that  $\Pi_{\theta_r^*}(\mathbf{x}) := \hat{\mathbf{x}}$  is a continuity point of  $u_0$ . Note that

$$\begin{aligned} \|u_h(\mathbf{x}) - u_0(\hat{\mathbf{x}})\| &\leq \|u_h(\mathbf{x}) - (\tilde{g}_{\sigma(h)} * u_0)(\hat{\mathbf{x}})\| \\ &\quad + \|(\tilde{g}_{\sigma(h)} * u_0)(\hat{\mathbf{x}}) - u_0(\hat{\mathbf{x}})\|. \end{aligned}$$

By Theorem 2, for  $h$  sufficiently small the first term in the above sum is at most  $\frac{\epsilon}{2}$ . It suffices to show that taking  $h$  sufficiently small bounds the second term by  $\frac{\epsilon}{2}$  as well.

For convenience, within this proof given  $\mathbf{x} := (x_1, x_2)$ ,  $\mathbf{y} := (y_1, y_2) \in [0, 1) \times \mathbb{R}$  we define  $B_{r,h}(\mathbf{x})$  and  $\mathbf{x} - \mathbf{y}$  in the circular sense, that is

$$B_{r,h}(\mathbf{x}) := \{\mathbf{y} : d_1^{\text{circ}}(x_1, y_1)^2 + (x_2 - y_2)^2 \leq r^2\}$$

$$\mathbf{x} - \mathbf{y} := (x_1 - y_1 \mod 1, x_2 - y_2).$$

For any  $r > 0$  we have

$$\begin{aligned} & \|(\tilde{g}_{\sigma(y,h)} * u_0)(\hat{\mathbf{x}}) - u_0(\hat{\mathbf{x}})\| \\ & \leq \left\| \sum_{\mathbf{y} \in \mathcal{U}_h \cap B_{r,h}(\hat{\mathbf{x}})} \tilde{g}_{\sigma(y,h)}(\hat{\mathbf{x}} - \mathbf{y}) u_0(\mathbf{y}) - u_0(\hat{\mathbf{x}}) \right\| \\ & \quad + \left\| \sum_{\mathbf{y} \in \mathcal{U}_h \setminus B_{r,h}(\hat{\mathbf{x}})} \tilde{g}_{\sigma(h)}(\hat{\mathbf{x}} - \mathbf{y}) u_0(\mathbf{y}) - u_0(\hat{\mathbf{x}}) \right\| \\ & \leq \sum_{\mathbf{y} \in \mathcal{U}_h \cap B_{r,h}(\hat{\mathbf{x}})} \tilde{g}_{\sigma(h)}(\hat{\mathbf{x}} - \mathbf{y}) \|u_0(\mathbf{y}) - u_0(\hat{\mathbf{x}})\| \\ & \quad + \sum_{\mathbf{y} \in \mathcal{U}_h \setminus B_{r,h}(\hat{\mathbf{x}})} \tilde{g}_{\sigma(h)}(\hat{\mathbf{x}} - \mathbf{y}) \|u_0(\mathbf{y}) - u_0(\hat{\mathbf{x}})\|. \end{aligned}$$

Take  $r$  small enough that  $\mathbf{y} \in B_{r,h}(\hat{\mathbf{x}})$  implies  $|u_0(\mathbf{y}) - u_0(\hat{\mathbf{x}})| < \frac{\epsilon}{4}$ . At the same time, since  $\sigma(y, h) \rightarrow 0$  as  $h \rightarrow 0$ , taking  $h$  sufficiently small ensures that the mass of  $\tilde{g}_{\sigma(y,h)}(\hat{\mathbf{x}} - \mathbf{y})$  outside of  $B_{r,h}(\hat{\mathbf{x}})$  is no more than  $\frac{\epsilon}{4M}$ . This gives

$$\begin{aligned} \|(\tilde{g}_{\sigma(y,h)} * u_0)(\hat{\mathbf{x}}) - u_0(\hat{\mathbf{x}})\| & \leq \sum_{\mathbf{y} \in \mathcal{U}_h \cap B_{r,h}(\hat{\mathbf{x}})} \tilde{g}_{\sigma(h)}(\hat{\mathbf{x}} - \mathbf{y}) \frac{\epsilon}{4} \\ & \quad + \sum_{\mathbf{y} \in \mathcal{U}_h \setminus B_{r,h}(\hat{\mathbf{x}})} \tilde{g}_{\sigma(h)}(\hat{\mathbf{x}} - \mathbf{y}) M \\ & \leq 1 \cdot \frac{\epsilon}{4} + \frac{\epsilon}{4M} \cdot M \leq \frac{\epsilon}{2} \end{aligned}$$

as required.  $\square$

**Remark 13** Here we have derived the fixed-ratio continuum limit as a corollary of the asymptotic limit. This was done for the sake of efficiency - it can also be derived as a stand alone result. Moreover, this result is much more general, covering convergence in  $L^p$  rather than just pointwise convergence, and the assumption that the stencil  $a_r^*$  not be restricted to any sublattice of  $\mathbb{Z}^2$  (Definition 3) can be eliminated. Convergence rates are also given, which turn out to depend on both the choice of  $L^p$  norm and the regularity of the boundary data  $u_0$ . However, the proof is very long and technical, so for this reason we have published it online only—please see [26].

## 7 Consequences

In this section apply the fixed-ratio continuum limit and asymptotic limit of the previous section, that is Theorems 2 and 6, in order to explain:

1. The “kinking” and “blur” artifacts listed in Sect. 2.2 and illustrated in Figs. 5, 6, and 8.
2. The differences between the direct form of Algorithm 1 and its semi-implicit extension in terms of coping with these artifacts.

We begin by considering kinking artifacts, and then go on to blur. We will see that the fixed-ratio continuum limit from Theorem 6 is all that is required to understand kinking artifacts, while blur artifacts require the asymptotic limit from Theorem 2. Shocks and cut off isophotes are not considered as they have already been adequately explained elsewhere [31,32].

### 7.1 Kinking

We begin by exploring kinking artifacts. First we prove a fundamental distinction between the direct form of Algorithm 1 and the semi-implicit extension. Then we go on to look at the limiting transport directions associated with three specific methods: coherence transport, Guidefill, and semi-implicit Guidefill.

#### 7.1.1 The Direct form Algorithm 1 Kinks, the Semi-Implicit Extension Need Not

An immediate corollary of Theorem 6 is that the direct form of Algorithm 1 is limited in terms of what directions it can extrapolate along, while this is not true of the semi-implicit extension. This follows from the geometric interpretation of the limiting transport direction  $\mathbf{g}_r^*$  as the center of mass of the stencil  $a_r^*$  with the respect to the stencil weights  $\{w_r(\mathbf{0}, \mathbf{y})/W : \mathbf{y} \in a_r^*\}$ . Specifically, since the original weights  $\{\frac{w_r(\mathbf{0}, \mathbf{y})}{W}\}_{\mathbf{y} \in a_r^*}$  are non-negative and sum to one, by the preservation of total mass (10) and inheritance of non-negativity (12) properties of equivalent weights, the same is true of the equivalent weights  $\{\frac{\tilde{w}_r(\mathbf{0}, \mathbf{y})}{W}\}_{\mathbf{y} \in \text{Supp}(a_r^*)}$ . Hence

$$\mathbf{g}_r^* = \sum_{\mathbf{y} \in \text{Supp}(a_r^*)} \frac{\tilde{w}_r(\mathbf{0}, \mathbf{y})}{W} \mathbf{y} \in \text{Conv}(\text{Supp}(a_r^*)),$$

where  $\text{Conv}(\text{Supp}(a_r^*))$  denotes the convex hull of  $\text{Supp}(a_r^*)$ . But by (26) we have

$$\text{Supp}(a_r^*) \subseteq \begin{cases} \bar{b}_r^- & \text{direct form of Algorithm 1} \\ \bar{b}_r^0 & \text{semi-implicit form} \end{cases} \quad (56)$$

where  $\bar{b}_r^0$  and  $\bar{b}_r^-$  are the dilated sets defined by (24) and (25) respectively. It follows that  $\mathbf{g}_r^*$  has to lie in the convex hull of  $\bar{b}_r^-$  if the direct form of Algorithm 1 is used, or the convex hull of  $\bar{b}_r^0$  if the semi-implicit form is used instead. That is

$$\mathbf{g}_r^* \in \begin{cases} \text{Conv}(\bar{b}_r^-) & \text{direct form of Algorithm 1.} \\ \text{Conv}(\bar{b}_r^0) & \text{semi-implicit form.} \end{cases} \quad (57)$$

This in turn immediately implies that if we use the direct form of Algorithm 1, then  $\theta(\mathbf{g}_r^*)$  is restricted to the cone

$$\theta_c \leq \theta(\mathbf{g}_r^*) \leq \pi - \theta_c \quad (58)$$

where

$$\theta_c = \arcsin\left(\frac{1}{r}\right) \quad (59)$$

is the smallest possible angle one can make given the restriction (57), while for the semi-implicit method we have

$$0 < \theta(\mathbf{g}_r^*) < \pi \quad (60)$$

where the angles  $\theta = 0$  and  $\theta = \pi$  are omitted not because of the restriction (57), but because Theorem 6 does not apply in either case (indeed, the continuum limit  $u$  given by (30) is not even defined). The cone (58) is exactly what we saw in Fig. 5c. At the same time, the lack of restrictions implied by (60) is consistent with our experience in Figs. 12 and 14, where semi-implicit Guidefill was able to successfully extrapolate lines making an angle as small as  $1^\circ$  with the inpainting domain boundary for  $r = 3$ , well under the critical angle  $\theta_c = \arcsin(\frac{1}{3}) \approx 19.5^\circ$  where standard Guidefill breaks down. Figure 16 illustrates this result.

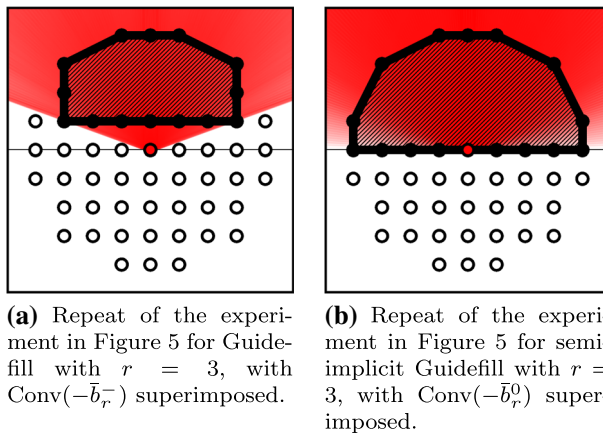
### 7.1.2 Limiting Transport Directions for Coherence Transport, Guidefill, and Semi-Implicit Guidefill

Here we derive formulas for the limiting transport directions of coherence transport, Guidefill, and semi-implicit Guidefill in the limit as  $\mu \rightarrow \infty$ . For convenience, in this section we rescale the limiting transport direction  $\mathbf{g}_r^*$  by a factor of  $W = \sum_{\mathbf{y} \in a_r^*} w_r(\mathbf{0}, \mathbf{y})$ , giving

$$\mathbf{g}_r^* = \sum_{\mathbf{y} \in a_r^*} w_r(\mathbf{0}, \mathbf{y}) \mathbf{y}. \quad (61)$$

This is more convenient to work with and defines the same transport equation. In fact, the ability to rescale  $\mathbf{g}_r^*$  by any  $\lambda \neq 0$  without changing the underlying transport equation is a tool that we will use repeatedly in our arguments throughout this section. To make the dependence of the transport direction on  $\mu$  explicit, within this section we write this direction as  $\mathbf{g}_{r,\mu}^*$ , and define

$$\mathbf{g}_r^* := \lim_{\mu \rightarrow \infty} \mathbf{g}_{r,\mu}^*.$$



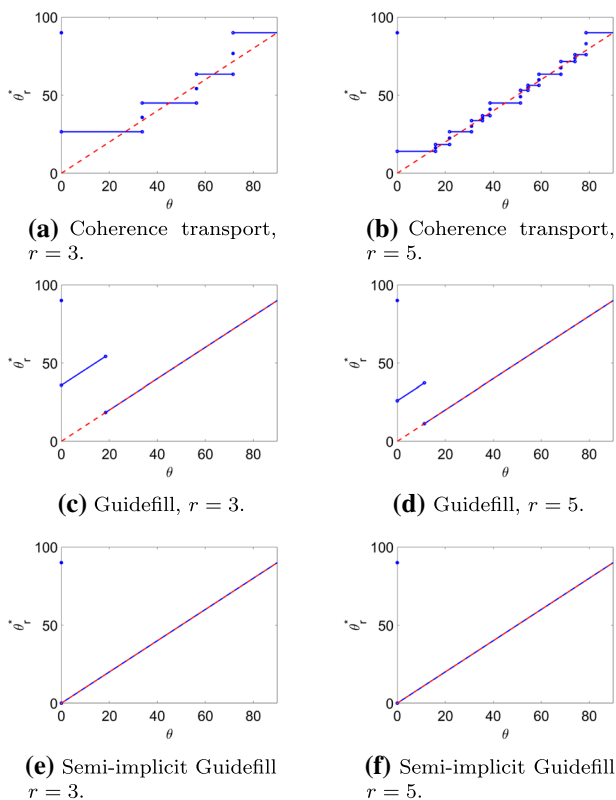
**Fig. 16** Convex hulls and transport directions: An immediate corollary of Theorem 6 is that the limiting transport direction  $\mathbf{g}_r^*$  lies in the convex hull of  $\bar{b}_r^-$  for the direct form of Algorithm 1, and the convex hull of  $\bar{b}_r^0$  for the semi-implicit extension. Since  $\text{Conv}(\bar{b}_r^-)$  contains only a cone of directions while  $\text{Conv}(\bar{b}_r^0)$  contains the full arc from 0 to  $\pi$ , this explains why Guidefill (and more generally *all* direct methods of the general form given by Algorithm 1) fail for shallow angles, while this is not true of the semi-implicit extension. To illustrate this, we have repeated the experiment from Fig. 5 using Guidefill (a) and semi-implicit Guidefill (b), while superimposing the sets  $\text{Conv}(-\bar{b}_r^-)$  and  $\text{Conv}(-\bar{b}_r^0)$  respectively (we have negated the sets for convenience which we can do since  $\mathbf{g}_r^*$  and  $-\mathbf{g}_r^*$  define the same transport equation). Note that in the case of semi-implicit Guidefill, the lines appear to be getting fainter as  $\theta(\mathbf{g}_r^*) \rightarrow 0$  and  $\theta(\mathbf{g}_r^*) \rightarrow \pi$ . This likely has two causes. For one, the angular footprint of the red dot in Fig. 5a, i.e. the width of the dot multiplied by  $\sin \theta_r^*$  (which tells you how much of the dot is “visible” from direction  $\theta_r^*$ ) is going to zero. Secondly, we will see in Sect. 7.3 (see in particular Fig. 23) that semi-implicit Guidefill has blur artifacts that become arbitrarily bad as  $\theta(\mathbf{g}_r^*) \rightarrow 0$  or  $\theta(\mathbf{g}_r^*) \rightarrow \pi$ . Nevertheless, Fig. 12 demonstrates that semi-implicit Guidefill can successfully extrapolate lines with  $\theta$  very close to 0 without serious issues of faintness. All parameters are the same as in Fig. 5

In order to resolve the ambiguity that  $\mathbf{g}_r^*$  and  $-\mathbf{g}_r^*$  define the same transport equation, we will frequently be taking angles modulo  $\pi$ . This is reflected in the definition we have selected for  $\theta(\mathbf{v})$ —recall that this refers to the angle that the line  $L_{\mathbf{v}} = \{\lambda \mathbf{v} : \lambda \in \mathbb{R}\}$  makes with the  $x$ -axis, and hence always lies in the range  $[0, \pi)$ . We define  $\mathbf{g} = (\cos \theta, \sin \theta)$ ,  $\theta_{r,\mu}^* := \theta(\mathbf{g}_{r,\mu}^*)$ ,  $\theta_r^* := \theta(\mathbf{g}_r^*)$  and consider the function  $\theta_r^* = F(\theta)$ . Our main concern is to determine when  $\theta_r^* = \theta$  (no kinking) and when  $\theta_r^* \neq \theta$  (kinking). Results are illustrated in Fig. 17.

**Coherence transport** We have already stated in Sect. 2 and illustrated in Fig. 5 that coherence transport in the limit  $\mu \rightarrow \infty$  kinks unless  $\mathbf{g} = \lambda \mathbf{v}$  for some  $\mathbf{v} \in B_{\epsilon,h}(\mathbf{0})$  and some  $\lambda \in \mathbb{R}$ . Under the assumptions of Sect. 6.1, a more precise statement is that coherence transport in the limit  $\mu \rightarrow \infty$  fails to kink if and only if  $\mathbf{g} = \lambda \mathbf{v}$  for some  $\mathbf{v} \in b_r^-$  and  $\lambda \in \mathbb{R}$ . Before we prove this, let us make some definitions. We define the *angular spectrum* of  $b_r^-$  as

$$\Theta(b_r^-) := \{\theta_1, \dots, \theta_n \in [0, \pi) : \exists \mathbf{y} \in b_r^- \text{ s.t. } \theta_i = \theta(\mathbf{y})\}. \quad (62)$$

In other words,  $\Theta(b_r^-)$  is the collection of angles modulo  $\pi$  that are representable using members of  $b_r^-$  (or which elements of the projective space  $\mathbb{RP}^1$  are representable, to be more mathematically precise), see Fig. 18 for an illustration. The angular spectrum



**Fig. 17** Limiting transport direction  $\theta_r^* = \theta(\mathbf{g}_r^*)$  as a function of the guidance direction  $\theta = \theta(\mathbf{g})$  for the three main methods: The theoretical limiting curves  $\theta_r^* = F(\theta)$  obtained for coherence transport (a), (b), Guidefill (c), (d), and semi-implicit Guidefill (e), (f). The desired curve  $F(\theta) = \theta$  is highlighted in red. Coherence transport exhibits a staircase pattern with  $F(\theta) \neq \theta$  for all but finitely many  $\theta$ , Guidefill obeys  $F(\theta) = \theta$  for all  $\theta \in (\theta_c, \pi - \theta_c)$  where  $\theta_c$  is the critical angle (59), and semi-implicit Guidefill obeys  $F(\theta) = \theta$  for all  $\theta \neq 0$ . All angles are in degrees

may be similarly defined in the obvious way for more general sets, and we do so in “Appendix E”. We will show that for coherence transport, when  $0 < \theta < \pi$ , we either have

$$\theta_r^* \in \Theta(b_r^-)$$

or

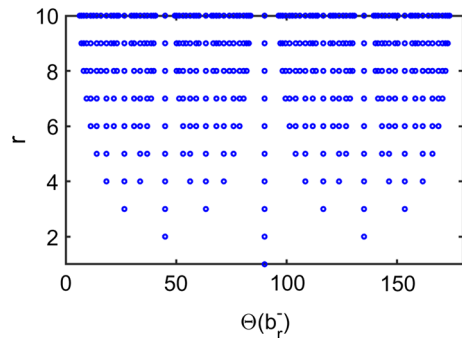
$$\theta_r^* = \frac{\theta_i + \theta_{i+1}}{2} \quad \text{for consecutive } \theta_i, \theta_{i+1} \in \Theta(b_r^-).$$

To begin, note that in this case (61) becomes

$$\mathbf{g}_{r,\mu}^* := \sum_{\mathbf{y} \in b_r^-} e^{-\frac{\mu^2}{2r^2}(\mathbf{y} \cdot \mathbf{g}^\perp)^2} \frac{\mathbf{y}}{\|\mathbf{y}\|},$$



**Fig. 18** Illustration of the angular spectrum  $\Theta(b_r^-)$  tells us which angles (modulo  $\pi$ ), are representable using elements of  $b_r^-$ . Here we have illustrated  $\Theta(b_r^-)$ , measured in degrees, for  $r = 1, 2, \dots, 10$



where  $b_r^-$  is the discrete half ball (23). Denote by  $\Psi$  the set of minimizers of  $|\mathbf{y} \cdot \mathbf{g}^\perp|$  for  $\mathbf{y} \in b_r^-$ , meaning that  $|\mathbf{y} \cdot \mathbf{g}^\perp| := \Delta \geq 0$  for all  $\mathbf{y} \in \Psi$  and  $|\mathbf{y} \cdot \mathbf{g}^\perp| > \Delta$  for all  $\mathbf{y} \in b_r^- \setminus \Psi$ . After rescaling by  $e^{\frac{\mu^2}{2r^2} \Delta^2}$ , the transport direction  $\mathbf{g}_{r,\mu}^*$  becomes

$$\begin{aligned} \mathbf{g}_{r,\mu}^* &= \sum_{\mathbf{y} \in \Psi} \frac{\mathbf{y}}{\|\mathbf{y}\|} + \sum_{\mathbf{j} \in b_r^- \setminus \Psi} e^{-\frac{\mu^2}{2r^2} \{(\mathbf{y} \cdot \mathbf{g}^\perp)^2 - \Delta^2\}} \frac{\mathbf{y}}{\|\mathbf{y}\|} \\ &\rightarrow \sum_{\mathbf{y} \in \Psi} \frac{\mathbf{y}}{\|\mathbf{y}\|} \quad \text{as } \mu \rightarrow \infty. \end{aligned} \quad (63)$$

Note that  $|\mathbf{y} \cdot \mathbf{g}^\perp|$  represents the distance from the point  $\mathbf{y}$  to the line through the origin  $L_{\mathbf{g}}$ . Thus computing the set  $\Psi$  is equivalent to finding the set of points in  $b_r^-$  closest to a given line through the origin. In “Appendix E” we prove that as  $\theta$  sweeps out an arc from 0 to  $\pi$ , for all but finitely many  $\theta$  the set  $\Psi$  is a singleton, containing a sequence of lone minimizers that we enumerate (in order of occurrence) as  $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{n'}$  (for some finite  $n'$ ). Now, it turns out that  $n' = n$  and moreover for every  $\theta_i \in \Theta(b_r^-)$  we have

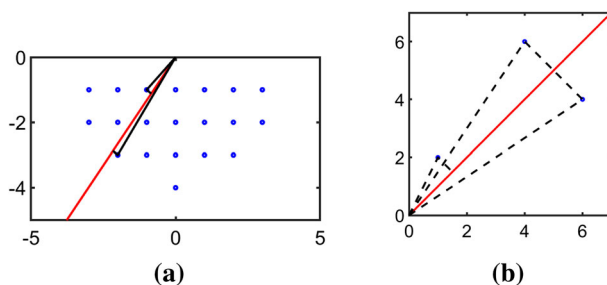
$$\theta_i = \theta(\mathbf{y}_i)$$

(“Appendix E”, Proposition 4). In other words, we have a 1-1 correspondence between (singleton) minimizers of  $|\mathbf{y} \cdot \mathbf{g}^\perp|$  and the angular spectrum  $\Theta(b_r^-)$ . Moreover, it can be shown that if  $\theta_i < \theta < \theta_{i+1}$  for some  $\theta_i, \theta_{i+1} \in \Theta(b_r^-)$ , then either  $\Psi = \{\mathbf{y}_i\}$  (for  $\theta$  close to  $\theta_i$ ) or  $\Psi = \{\mathbf{y}_{i+1}\}$  (for  $\theta$  close to  $\theta_{i+1}$ ) or  $\Psi = \{\mathbf{y}_i, \mathbf{y}_{i+1}\}$  if  $\theta = \theta_{i,i+1}$ , where  $\theta_{i,i+1} \in (\theta_i, \theta_{i+1})$  is a critical angle we refer to as a *transition angle*.

**Definition 7** Given minimizers  $\{\mathbf{y}_i\}_{i=1}^n$  defined as above we define the set of *transition angles*  $\{\theta_{i,i+1}\}_{i=1}^{n-1}$  by

$$\theta_{i,i+1} = \theta(\mathbf{y}_i + \mathbf{y}_{i+1}) \quad \text{for } 1 \leq i \leq n-1,$$

where  $\theta(\mathbf{y})$  denotes as usual the angle between the line  $L_{\mathbf{y}}$  and the x-axis.



**Fig. 19** Closest points and shallowest angles: we claim that the closest point in the set  $b_r^- \subset \mathbb{Z}^2$  to a given line  $L_{\mathbf{g}}$  ( $\mathbf{g} = (\cos \theta, \sin \theta)$ ) is always one of the two points casting the shallowest angle with  $L_{\mathbf{g}}$  on either side, as illustrated in **a** for  $\theta = 53^\circ$ ,  $r = 4$ . This statement does *not* hold if  $b_r^-$  is replaced with a generic  $A \subset \mathbb{Z}^2$ , as demonstrated by the counterexample  $A = \{(1, 2), (4, 6), (6, 4)\}$  and  $\theta = 45^\circ$  in **b**

For convenience, we also define  $\theta_{0,1} = 0$  and  $\theta_{n,n+1} = \pi$ . Since one can also prove  $\Psi = \{\mathbf{y}_1\}$  for  $0 := \theta_{0,1} < \theta < \theta_1$  and  $\Psi = \{\mathbf{y}_n\}$  for  $\theta_n < \theta < \theta_{n,n+1} := \pi$ , with this notation we have the general result

$$\Psi = \begin{cases} \{\mathbf{y}_i\} & \text{if } \theta_i < \theta < \theta_{i,i+1} \\ & \text{for some } i = 1, \dots, n \\ \{\mathbf{y}_i, \mathbf{y}_{i+1}\} & \text{if } \theta = \theta_{i,i+1} \\ & \text{for some } i = 1, \dots, n-1 \\ \{\mathbf{y}_{i+1}\} & \text{if } \theta_{i,i+1} < \theta < \theta_{i+1} \\ & \text{for some } i = 0, \dots, n-1 \end{cases}$$

(“Appendix E”, Proposition 5—note that we have carefully excluded  $\theta_{0,1} := 0$  and  $\theta_{n,n+1} = \pi$  from the middle case). In words, this means that the element(s) of  $b_r^-$  closest to the line  $L_{\mathbf{g}}$  are also the member(s) of  $b_r^-$  that cast the shallowest angles with  $L_{\mathbf{g}}$  from above and below. This statement is not true if  $b_r^-$  is replaced with a generic subset of  $\mathbb{Z}^2$ —see Fig. 19. The remaining cases are  $\theta = \theta_i \in \Theta(b_r^-)$  and  $\theta = 0$ . We deal with the former first—in the first case we have

$$\Psi = \{\mathbf{y} \in b_r^- : \theta(\mathbf{y}) = \theta_i\},$$

a set containing up to  $r$  members, all of which are parallel to each other and to  $\mathbf{g}$ . In order to make these ideas more concrete, Example 1 gives explicit expressions for  $\Theta(b_r^-)$  and  $\Psi$  in the case  $r = 3$ . Also, in “Appendix E” Remark 20, we give an algorithm for computing  $\Theta(b_r^-)$  and  $Y(b_r^-) := \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n\}$  for any  $r$ .

**Example 1** When  $r = 3$  we have

$$\begin{aligned} \Theta(b_3^-) &:= \{\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7\} \\ &= \{\arctan(1/2), \frac{\pi}{4}, \arctan(2), \frac{\pi}{2}, \\ &\quad \frac{\pi}{2} + \arctan(1/2), \frac{3\pi}{4}, \frac{\pi}{2} + \arctan(2)\}. \end{aligned}$$

For  $0 < \theta \leq \frac{\pi}{2}$ , (we omit  $\frac{\pi}{2} < \theta < \pi$  for brevity) the set of minimizers  $\Psi$  is given by

$$\Psi = \begin{cases} \{(-2, -1)\} := \{\mathbf{y}_1\} \\ \text{if } 0 < \theta < \theta_{1,2}. \\ \{(-2, -1), (-1, -1)\} := \{\mathbf{y}_1, \mathbf{y}_2\} \\ \text{if } \theta = \theta_{1,2}. \\ \{(-1, -1)\} := \{\mathbf{y}_2\} \\ \text{if } \theta_{1,2} < \theta < \theta_2. \\ \{(-1, -1), (-2, -2)\} := \{\mathbf{y}_2, 2\mathbf{y}_2\} \\ \text{if } \theta = \theta_2. \\ \{(-1, -1)\} := \{\mathbf{y}_2\} \\ \text{if } \theta_2 < \theta < \theta_{2,3}. \\ \{(-1, -1), (-1, -2)\} := \{\mathbf{y}_2, \mathbf{y}_3\} \\ \text{if } \theta = \theta_{2,3}. \\ \{(-1, -2)\} = \{\mathbf{y}_3\} \\ \text{if } \theta_{2,3} < \theta < \theta_{3,4}. \\ \{(-1, -2), (0, -1)\} = \{\mathbf{y}_3, \mathbf{y}_4\} \\ \text{if } \theta = \theta_{3,4}. \\ \{(0, -1)\} := \{\mathbf{y}_4\} \\ \text{if } \theta_{3,4} < \theta < \theta_4. \\ \{(0, -1), (0, -2), (0, -3)\} := \{\mathbf{y}_4, 2\mathbf{y}_4, 3\mathbf{y}_4\} \\ \text{if } \theta = \theta_4. \end{cases}$$

where  $\theta_{0,1} = 0$ ,  $\theta_{1,2} = \arctan(2/3)$ ,  $\theta_{2,3} = \arctan(3/2)$ ,  $\theta_{3,4} = \arctan(3)$ .

When  $\Psi$  is a singleton set, that is  $\Psi = \{\mathbf{y}_i\}$  for some  $1 \leq i \leq n$ , (63) becomes  $\mathbf{g}_r^* = \frac{\mathbf{y}_i}{\|\mathbf{y}_i\|}$  and we have

$$\theta(\mathbf{g}_r^*) = \theta\left(\frac{\mathbf{y}_i}{\|\mathbf{y}_i\|}\right) = \theta_i$$

On the other hand, if  $\theta = \theta_i \in \Theta(b_r^-)$ ,  $\mathbf{g}_r^*$  is a sum of vectors all parallel to one another and to  $\mathbf{g}$ , and we get  $\theta_i^* = \theta_i$  again. This is the lone case in which coherence transport doesn't kink. Next, at the transition angles  $\theta_{i,i+1}$  where  $\Psi = \{\mathbf{y}_i, \mathbf{y}_{i+1}\}$ , we have  $\mathbf{g}_r^* = \frac{\mathbf{y}_i}{\|\mathbf{y}_i\|} + \frac{\mathbf{y}_{i+1}}{\|\mathbf{y}_{i+1}\|}$ , so that

$$\theta(\mathbf{g}_r^*) = \theta\left(\frac{\mathbf{y}_i}{\|\mathbf{y}_i\|} + \frac{\mathbf{y}_{i+1}}{\|\mathbf{y}_{i+1}\|}\right) = \frac{\theta_i + \theta_{i+1}}{2},$$

where we have used the observation (proved in “Appendix E”, Observation 7) that  $\theta(\mathbf{v} + \mathbf{w}) = \frac{\theta(\mathbf{v}) + \theta(\mathbf{w})}{2}$  holds for all unit vectors  $\mathbf{v}, \mathbf{w} \in S^1$ . Finally, suppose  $\theta = 0$ .

Here we have  $\Psi = \{(i, -1) : -r + 1 \leq i \leq r - 1\}$ , giving

$$\mathbf{g}_r^* = e_2 \quad \text{for } \theta = 0$$

after rescaling. In summary, for  $\theta \in [0, \pi)$  we then have

$$\theta_r^* = \begin{cases} \frac{\pi}{2} & \text{if } \theta = 0 \\ \theta_i & \text{if } \theta_{i-1,i} < \theta < \theta_{i,i+1} \text{ for some } i = 1, \dots, n \\ \frac{\theta_i + \theta_{i+1}}{2} & \text{if } \theta = \theta_{i,i+1} \text{ for some } i = 1, \dots, n \end{cases} \quad (64)$$

See Fig. 17a, b for an illustration of (64) for  $r = 3$  and  $r = 5$ . In “Appendix E”, Corollary 2 we prove a generalization of this result that applies if, for example, the discrete ball used by coherence transport is replaced with a discrete square.

**Guidefill.** In this case (61) becomes

$$\mathbf{g}_{r,\mu}^* := \sum_{\mathbf{y} \in \tilde{b}_r^-} e^{-\frac{\mu^2}{2r^2}(\mathbf{y} \cdot \mathbf{g}^\perp)^2} \frac{\mathbf{y}}{\|\mathbf{y}\|},$$

where  $\tilde{b}_r^-$  is given by (23). It is useful to partition  $\tilde{b}_r^-$  into a disjoint union of sets  $\ell_k^-$  such at each  $\ell_k^-$  is the collection of points in  $\tilde{b}_r^-$  distance  $k$  from the line  $L_{\mathbf{g}} = \{\lambda \mathbf{g} : \lambda \in \mathbb{R}\}$ , that is

$$\ell_k^- = \{n\mathbf{g} + m\mathbf{g}^\perp \in \tilde{b}_r^- : m = \pm k\}.$$

Since the weights (6) exponentially decay with distance from  $L_{\mathbf{g}}$ , then so long as  $\ell_0^-$  is non-empty, we expect the contribution of the other  $\ell_k^-$  to vanish. For Guidefill,  $\ell_0^-$  can be explicitly parametrized as

$$\begin{aligned} \ell_0^- &:= \{n\mathbf{g} \in \tilde{b}_r^- : n\mathbf{g} \cdot e_2 \leq -1\} \\ &= \{n\mathbf{g} : n = -r, \dots, -\lceil \csc \theta \rceil\}. \end{aligned}$$

For  $\ell_0^-$  to be non-empty, we need  $\lceil \csc \theta \rceil \leq r$ , which occurs only if  $\theta_c \leq \theta \leq \pi - \theta_c$ , where  $\theta_c$  is the same critical angle (59) from Sect. 7.1.1. If  $\ell_0^- \neq \emptyset$ , we have

$$\begin{aligned} \mathbf{g}_{r,\mu}^* &= \sum_{\mathbf{y} \in \ell_0^-} \frac{\mathbf{y}}{\|\mathbf{y}\|} + \sum_{k=1}^r \sum_{\mathbf{y} \in \ell_k^-} e^{-\frac{\mu^2}{2r^2} k^2 \|\mathbf{g}\|^2} \frac{\mathbf{y}}{\|\mathbf{y}\|} \\ &\rightarrow \sum_{\mathbf{y} \in \ell_0^-} \frac{\mathbf{y}}{\|\mathbf{y}\|} \quad \text{as } \mu \rightarrow \infty. \\ &= \sum_{n=-r}^{-\lceil \csc \theta \rceil} n\mathbf{g} \\ &= \mathbf{g} \quad \text{after rescaling.} \end{aligned}$$

Hence we transport in the correct direction in this case.

One the other hand, if  $\ell_0^- = \emptyset$  but  $\theta \neq 0$ , then the weights (6) concentrate all their mass into  $\ell_1^-$ , with all other  $\ell_k^-$  vanishing. Unfortunately, unlike  $\ell^0$ , the set  $\ell_1^-$  is not parallel to  $\mathbf{g}$  so we expect kinking in this case. In general  $\ell_1^-$  can consist of two parallel components on either side of  $\ell_0^-$ . But in this case, since  $\ell_0^-$  lies entirely above the line  $y = -1$ , we know  $\ell_1^-$  consists of just one component and we can write it explicitly as

$$\ell_1^- = \{n\mathbf{g} - \text{sgn}(\cos \theta)\mathbf{g}^\perp : n = -r + 1, \dots, -1\}$$

(remember that  $\mathbf{g}^\perp$  denotes the *counterclockwise* rotation of  $\mathbf{g}$  by  $90^\circ$ ). After rescaling by  $e^{\frac{\mu^2}{2r^2}\|\mathbf{g}\|^2}$  we obtain

$$\begin{aligned} \mathbf{g}_{r,\mu}^* &= \sum_{\mathbf{y} \in \ell_1^-} \frac{\mathbf{y}}{\|\mathbf{y}\|} + \sum_{k=2}^r \sum_{\mathbf{y} \in \ell_k^-} e^{-\frac{\mu^2}{2r^2}(k^2-1)\|\mathbf{g}\|^2} \frac{\mathbf{y}}{\|\mathbf{y}\|} \\ &\rightarrow \sum_{\mathbf{y} \in \ell_1^-} \frac{\mathbf{y}}{\|\mathbf{y}\|} \quad \text{as } \mu \rightarrow \infty. \\ &= \left( \sum_{n=-r+1}^{-1} \frac{n}{\sqrt{1+n^2}} \right) \mathbf{g} \\ &\quad - \text{sgn}(\cos \theta) \left( \sum_{n=-r+1}^{-1} \frac{1}{\sqrt{1+n^2}} \right) \mathbf{g}^\perp \\ &= \mathbf{g} + \text{sgn}(\cos \theta) \alpha_r \mathbf{g}^\perp \quad \text{after rescaling.} \end{aligned}$$

where

$$\alpha_r = \frac{\sum_{n=1}^{r-1} \frac{1}{\sqrt{1+n^2}}}{\sum_{n=1}^{r-1} \frac{n}{\sqrt{1+n^2}}}.$$

Finally, if  $\theta = 0$  we have  $\tilde{b}_r^- = b_r^-$  and we obtain  $\mathbf{g}_r^* = e_2$  as for coherence transport. Defining  $\Delta\theta_r = \arctan(\alpha_r)$ , for  $\theta \in [0, \pi)$  we obtain

$$\theta_r^* = \begin{cases} \frac{\pi}{2} & \text{if } \theta = 0 \\ \theta + \Delta\theta_r & \text{if } 0 < \theta < \theta_c \\ \theta & \text{if } \theta_c \leq \theta \leq \pi - \theta_c \\ \theta - \Delta\theta_r & \text{if } \pi - \theta_c < \theta < \pi. \end{cases} \quad (65)$$

In other words, aside from exceptional case  $\theta = 0$ , we have  $\theta_r^* = \theta$  for the “well behaved” range of values  $\theta_c \leq \theta \leq \pi - \theta_c$ , but  $\theta_r^*$  jumps by a constant angle  $\Delta\theta_r$  near  $\theta = 0$  and  $\theta = \pi$ . The first few values of  $\Delta\theta_r$  are  $\Delta\theta_3 \approx 35.8^\circ$ ,  $\Delta\theta_4 \approx 30.0^\circ$ ,  $\Delta\theta_5 \approx 25.9^\circ$ . See Fig. 17c, d for an illustration of (65) for  $r = 3$  and  $r = 5$ .

**Semi-implicit Guidefill.** The analysis of semi-implicit Guidefill is the same as for Guidefill except that the set  $\tilde{b}_r^-$  is replaced by  $\tilde{b}_r^0$ . Defining  $\ell_k^0$  as the collection of points in  $\tilde{b}_r^0$  distance  $k$  from the line  $L_{\mathbf{g}} = \{\lambda \mathbf{g} : \lambda \in \mathbb{R}\}$  as before, we find that in this case

$$\ell_0^0 := \{n\mathbf{g} \in \tilde{b}_r : n\mathbf{g} \cdot e_2 \leq 0\}$$

is *never* empty. In fact, for  $0 \leq \theta < \pi$  we have

$$\ell_0^0 = \begin{cases} \{n\mathbf{g} : -r \leq n \leq -1\} & \text{if } 0 < \theta < \pi \\ \{n\mathbf{g} : -r \leq n \leq r, n \neq 0\} & \text{if } \theta = 0. \end{cases}$$

If  $\theta > 0$ , we proceed as before and find

$$\begin{aligned} \mathbf{g}_{r,\mu}^* &= \sum_{\mathbf{y} \in \ell_0^0} \frac{\mathbf{y}}{\|\mathbf{y}\|} + \sum_{k=1}^r \sum_{\mathbf{y} \in \ell_k^0} e^{-\frac{\mu^2}{2r^2} k^2 \|\mathbf{g}\|^2} \frac{\mathbf{y}}{\|\mathbf{y}\|} \\ &\rightarrow \sum_{\mathbf{y} \in \ell_0^0} \frac{\mathbf{y}}{\|\mathbf{y}\|} \quad \text{as } \mu \rightarrow \infty. \\ &= \sum_{n=-r}^{-1} n\mathbf{g} \\ &= \mathbf{g} \quad \text{after rescaling.} \end{aligned}$$

However, if  $\theta = 0$ , this argument doesn't work because the elements of  $\ell_0^0$  all cancel each other out. In fact, in this case we have

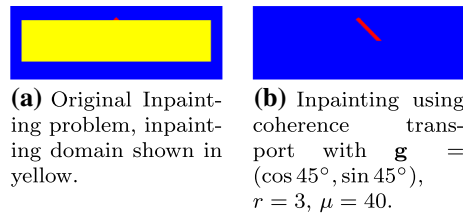
$$\mathbf{g}_{r,\mu}^* = \underbrace{\sum_{\mathbf{y} \in \ell_0^0} \frac{\mathbf{y}}{\|\mathbf{y}\|}}_{=0} + \underbrace{\sum_{\mathbf{y} \in \tilde{b}_r^-} e^{-\frac{\mu^2}{2r^2} (\mathbf{y} \cdot \mathbf{g}^\perp)^2} \frac{\mathbf{y}}{\|\mathbf{y}\|}}_{\text{the from coherence transport}} \mathbf{g}_{r,\mu}^*.$$

Hence, in this case  $\mathbf{g}_r^* = e_2$  yet again, just like for Guidefill and coherence transport. In general, for  $0 \leq \theta < \pi$ , we have

$$\theta_r^* = \begin{cases} \frac{\pi}{2} & \text{if } \theta = 0 \\ \theta & \text{if } 0 < \theta < \pi \end{cases} \quad (66)$$

In other words, semi-implicit Guidefill kinks only if  $\mathbf{g}$  is exactly parallel to boundary of the inpainting domain. See Fig. 17e, f for an illustration of (66) for  $r = 3$  and  $r = 5$  (the curves are of course the same, since (66) is independent of  $r$ ).

**Remark 14** In contrast to Guidefill and coherence transport, (66) tells us that for semi-implicit Guidefill in the limit  $\mu \rightarrow \infty$  we have  $\theta_r^* = \theta$  for all  $\theta \in (0, \pi)$ , independent of  $r$ . This is in fact exactly the same prediction (albeit under stronger simplifying



**Fig. 20** Stretching a dot into a line: In **a** we have an inpainting problem consisting of a red dot on a blue background, with the inpainting domain in yellow. In **b** we see the result of inpainting using coherence transport with  $\mu = 40$ ,  $r = 3$ , and  $\mathbf{g} = (\cos 45^\circ, \sin 45^\circ)$ . The dot is now stretched into a line, the orientation of which may be measured to deduce  $\mathbf{g}_r^*$

assumptions) that März and Bornemann obtained for coherence transport in their own continuum limit  $u_M$  as  $\mu \rightarrow \infty$  [7, p. 14]. We have in some sense come full circle—the original predictions of [7] for coherence transport under their high resolution and vanishing viscosity limit are the same as ours for semi-implicit Guidefill under our fixed ratio limit.

## 7.2 Numerical Validation of Limiting Transport Directions for Coherence Transport, Guidefill, and Semi-Implicit Guidefill

In this experiment we compare the limiting transport directions derived in Sect. 7.1.2 for coherence transport, Guidefill, and semi-implicit Guidefill as  $\mu \rightarrow \infty$  with the orientation of extrapolated isophotes in an actual inpainted image  $u_h$  obtained in practice with finite  $\mu$ . In each case we choose as our boundary data the image shown in Fig. 20a, consisting of a red dot on a blue background, with the inpainting domain shown in yellow. We run each algorithm with

$$\mathbf{g} = (\cos(k^\circ), \sin(k^\circ))$$

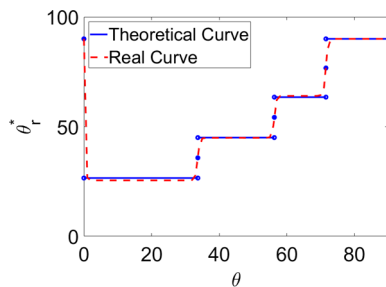
for  $k = 0, 1, \dots, 90$ , with  $\mu = 40$  fixed and for various values of  $r$ . The dot is then stretched into a line as in Fig. 20b, the orientation of which gives  $\mathbf{g}_{\mu,r}^*$  and which can be measured numerically.

Results are shown in Fig. 21 for  $r = 3$  (similar results may be found for different values of  $r$ , but these are omitted for brevity). The theoretical curves are shown in blue, while the measured curves are shown in red. While we see some smoothing out of the jump discontinuities in the case of coherence transport, this is expected as one can easily show that the convergence to  $\theta^* = \lim_{\mu \rightarrow \infty} \theta_{r,\mu}^*$  is pointwise but not uniform, becoming arbitrarily slow in the vicinity of the jumps. On the other hand, for Guidefill and semi-implicit Guidefill we see excellent agreement with theory even in the vicinity of jump discontinuities. This is to be expected as well, since one can easily show that the relevant limits are uniform in this case.

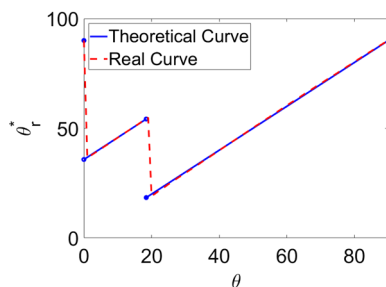
## 7.3 Consequences of the Asymptotic Limit

In this section, we apply the asymptotic limit from Theorem 2 to study the blur artifacts associated with Algorithm 1 and illustrated in Sect. 2.2. As a motivating example,

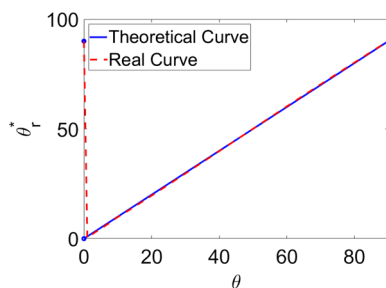
**Fig. 21** Validation of limiting transport directions for coherence transport, Guidefill, and semi-implicit Guidefill: here we compare the limiting transport directions  $\theta_r^* = \theta(\mathbf{g}_r^*)$  as a function of  $\theta = \theta(\mathbf{g})$  derived in Sect. 7.1.2 for coherence transport, Guidefill, and semi-implicit Guidefill as  $\mu \rightarrow \infty$  with the orientation of extrapolated isophotes in the inpainting problem shown in Fig. 20a (where  $\mu = 40$ ). The theoretical curves are shown in blue, while the measured curves are shown in red. In every case we have  $r = 3$



**(a)** Coherence transport,  $r = 3$ .



**(b)** Guidefill,  $r = 3$ .

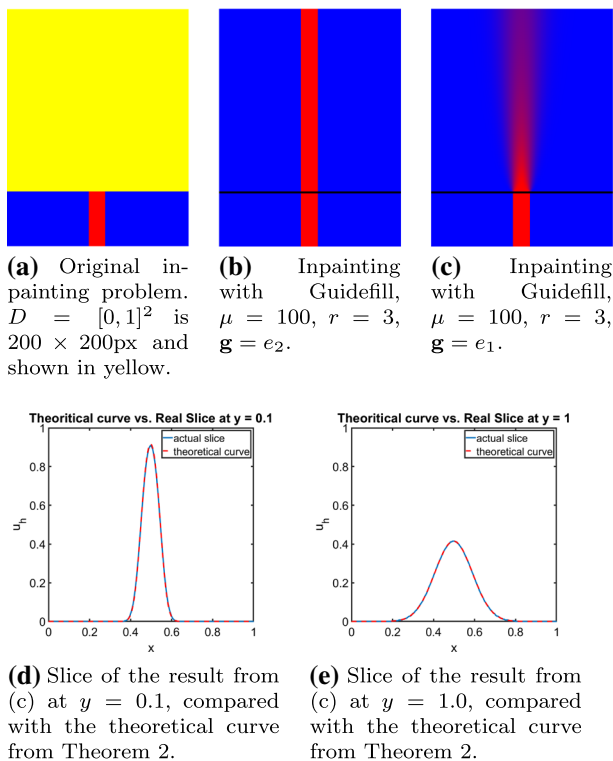


**(c)** Semi-implicit Guidefill,  $r = 3$ .

we consider the situation illustrated in Fig. 22, where we extrapolate a vertical line using Guidefill with two different guidance directions:  $\mathbf{g}_1 = e_2$  and  $\mathbf{g}_2 = e_1$ . We have already shown in Sect. 7.1.2 that these lead to the same continuum limit with transport direction  $\mathbf{g}_r^* = e_2$  (Fig. 17). However,  $\mathbf{g}_1 = e_2$  leads to a clean extrapolation (Fig. 22a) while the extrapolation using  $\mathbf{g}_1 = e_1$  suffers from heavy blur (Fig. 22b). Evidently, the fixed ratio continuum limit from Theorem 6 is inadequate to explain this discrepancy. Instead we turn to the asymptotic limit—plugging the relevant parameters into Theorem 2, we obtain a theoretical result that is indistinguishable from the result obtained in reality (Fig. 22c, d). The asymptotic limit thus appears to yield excellent predictions of the blur obtained in practice.

Although our asymptotic limit more closely approximates the real behavior of Algorithm 1 than the fixed-ratio continuum limit, the latter remains valid in the high resolution limit  $h \rightarrow 0$ . This is illustrated in Fig. 24 where we repeatedly solve the



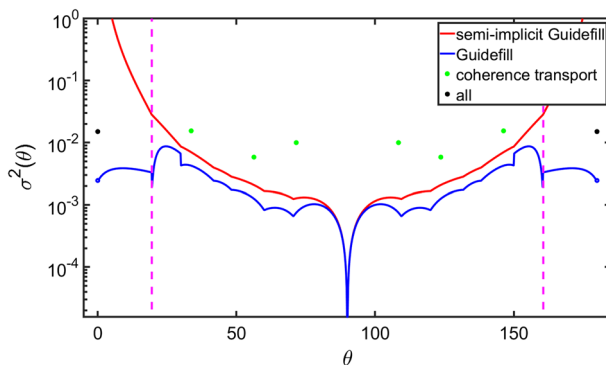


**Fig. 22** Transport is not the whole story: in this experiment, the problem shown in **a** of inpainting  $D = [0, 1]^2$  ( $200 \times 200$ px) given data on  $[0, 1] \times [-0.3, 0)$  is solved using Guidefill with  $\mu = 100$ ,  $r = 3$  and  $\mathbf{g} = (\cos \theta, \sin \theta)$  for  $\theta = \frac{\pi}{2}$  (**b**) and  $\theta = 0$  (**c**). While Sect. 7.1.2 tells us that  $\theta = 0$  and  $\theta = \frac{\pi}{2}$  have the same continuum limit, evidently they are very different in reality. This is reflected in the asymptotic limit (Theorem 2) which is able to predict differences not reflected in the continuum limit. In **c**, **d** we compare horizontal slices of **c** at  $y = 0.1$  and  $y = 1$  respectively with the predictions of Theorem 2. In this case the predictions are accurate to within an error of  $1/255$ , the minimum increment of an image on our machine

same inpainting problem with  $D = [0, 1) \times (0, 0.5]$  but at increasing high resolutions. By examining sections of constant  $y$  for differing values of  $h$ , we see explicitly the convergence of the discrete solution  $u_h$  to the fixed-ratio continuum limit  $u$ .

**Remark 15** Since  $u_h$  converges to  $u$  obeying the transport PDE (30), Algorithm 1 may be viewed as a numerical method for solving (30). It is well known (see for example [28]) that numerical methods for transport equations tend to introduce blur, and therefore we should not be surprised that this is also the case of Algorithm 1. One way of studying this blur is the technique of *modified equations* [28, p. 117]. However, this is not the approach we have adopted in this paper.

**Angular dependence of blur artifacts** Here we briefly explore how the variance  $\sigma(y, h)^2$  depends on  $\mathbf{g} = (\cos \theta, \sin \theta)$  as  $\theta$  changes. Since  $\sigma(y, h)^2 = \frac{\gamma^2 y h}{|\mu_y^3|}$  can be quite complex in general we explore this numerically. Figure 23 illustrates the angular dependence of  $\sigma(y, h)$  as a function of  $\theta \in [0, \pi]$  with  $y = 1$  and  $h = 0.01$  fixed, for



**Fig. 23** Angular variation of blurring artifacts: here we plot the angular dependence of the variance  $\sigma^2(y, h)$  from the asymptotic blur kernel  $g_{\sigma(y, h)}$  in Theorem 2, for the three methods coherence transport, Guidefill, and semi-implicit Guidefill. We fix with  $r = 3$ , take  $\mu \rightarrow \infty$ , vary  $\mathbf{g} = (\cos \theta, \sin \theta)$ , and plot  $\sigma^2(y, h)$  as a function of  $\theta$ . We fix  $y = 1$  and  $h = 0.01$

the three main methods of interest—coherence transport, Guidefill, and semi-implicit Guidefill (note the log scale). In every case we have  $r = 3$  and  $\mu = 100$ .

Note that for semi-implicit Guidefill, since  $\mu_y \rightarrow 0$  as  $\theta \rightarrow 0$ , we expect blur artifacts to become arbitrarily bad as  $\theta \rightarrow 0$ , unless we also have  $\gamma^2 \rightarrow 0$  as  $\theta \rightarrow 0$  in such a way that the ratio remains bounded. In fact,

$$\gamma^2 = \text{Var}(\mu_x V_1 - \mu_y U_1) = \text{Var}[(\mu_x, \mu_y) \cdot (U_1, V_1)].$$

is the variance of the increments  $\mathbf{Z}_1 = (U_1, V_1)$  orthogonal to the mean  $(\mu_x, \mu_y)$ , which *does* go to zero as  $\theta \rightarrow 0$ . However, as Fig. 23 illustrates, it does not do so fast enough to compensate for the blow up of  $1/|\mu_y|^3$ . Indeed, numerical experiments (omitted for brevity) demonstrate that the blur artifacts for semi-implicit Guidefill do become arbitrarily bad as  $\theta \rightarrow 0$  (Fig. 24).

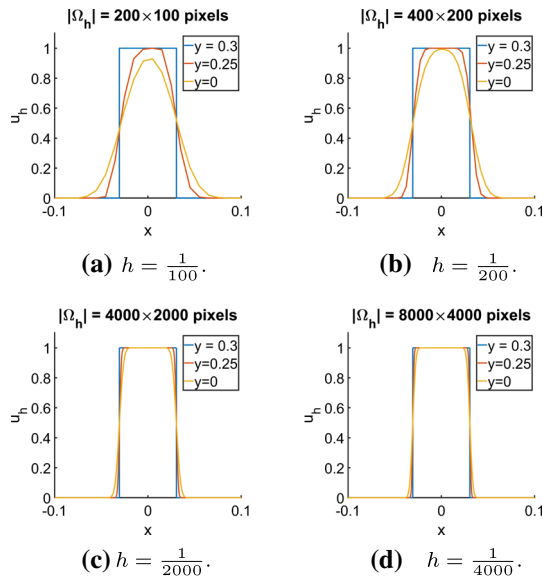
Note that for coherence transport,  $\sigma^2(y, h) = 0$  for all but finitely many angles, explaining the methods apparent lack of blur artifacts (Fig. 6). These special angles correspond precisely to the jumps in Fig. 17a, where coherence transport puts its mass into more than one  $\mathbf{y} \in b_r^-$  (Sect. 7.1.2).

**Degenerate stencils** As noted above, coherence transport suffers from no blur artifacts for all but finitely many angles. To understand why this is, recall from the analysis of Sect. 7.1.2 that coherence transport “kinks” because in the limit as  $\mu \rightarrow \infty$ , the stencil weights  $w_r(0, \mathbf{y})$  are zero for all but a single  $\mathbf{y}^* \in b_r^-$  which receives stencil weight one (except when  $\theta(\mathbf{g})$  is a transition angle or  $\theta(\mathbf{g}) = 0$ , in which case—as observed in numerical experiments omitted for brevity—coherence transport *does* blur). We call such stencils *degenerate*.

Theorem 2 cannot be applied to degenerate stencils as we have

$$\varphi_{Z_i}(\mathbf{u}) = e^{\sqrt{-1}\mathbf{y}^* \cdot \mathbf{u}}$$

which has magnitude one irrespective of  $\mathbf{u}$ . In this case, the “random” walk  $\mathbf{X}_\tau$  has become deterministic as the increments  $\Delta \mathbf{X}_i$  now obey  $\Delta \mathbf{X}_i = h \mathbf{Z}_i = h \mathbf{y}^*$  with



**Fig. 24** Convergence to the fixed-ratio continuum limit: the continuum problem of inpainting the line  $\tan(73^\circ) - 0.1 \leq y \leq \tan(73^\circ) + 0.1$  with image domain  $\Omega = [-1, 1] \times [-0.5, 0.5]$  and inpainting domain  $D = [-0.8, 0.8] \times [-0.3, 0.3]$  is rendered at a variety of resolutions and inpainted each time using Guidefill. Examining cross-sections of  $u_h$  at  $y = 0.3$  (on the boundary of  $D_h$ ),  $y = 0.25$  (just inside), and  $y = 0$  (in the middle of  $D_h$ ) we notice a gradual deterioration of the initially sharp signal. This deterioration is to be expected in light of Theorem 2, as  $u_h(\mathbf{x})$  is related to a mollified version of  $u_0$  with a Gaussian mollifier  $g_{\sigma(y,h)}$ . However, in light of Theorem 6, we also expect that as  $h \rightarrow 0$  and we approach the fixed-ratio continuum limit, this signal degradation should disappear. By examining the same slices at differing resolutions, we see that this is indeed the case

probability one. One may readily show that  $\rho_{\mathbf{x}_\tau}$  assigns mass 1 to a single pixel in  $\mathcal{U}_h$ , and hence there is no blur.

### Alternative interpolation schemes

As noted in Remark 7, if we were to modify semi-implicit Guidefill so that ghost pixels are defined not using bilinear interpolation, but by some other interpolation scheme that also

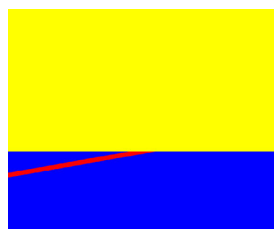
1. can be expressed in terms of non-negative basis functions summing to one
2. preserves polynomials of degree 1

we would obtain an identical *continuum limit* as for semi-implicit Guidefill. In the future, we would like to explore whether or not another interpolation scheme can be found such that  $\sigma(y, h)$  remains bounded as  $\theta \rightarrow 0$ . This would amount to finding a scheme for which the orthogonal variance  $\gamma^2 \rightarrow 0$  as  $\theta \rightarrow 0$  fast enough to compensate for the blow up of  $1/|\mu_y|^3$ .

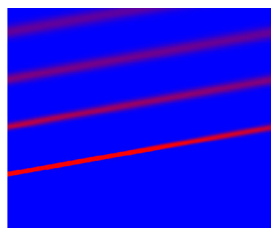
### Summary

Theorem 2 and the experiments of this section have four main takeaway messages:

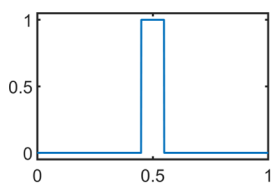
- Blur gets *worse* as one moves further into the inpainting domain and (e.g.  $y$  increases)—Figs. 22 and 25.



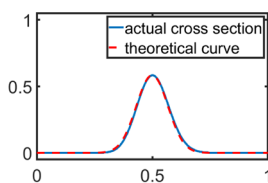
(a) Original inpainting problem ( $D_h$  in yellow).



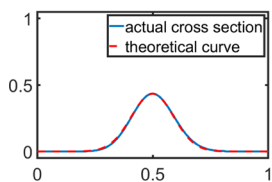
(b) Inpainted using semi-implicit Guidefill with periodic boundary conditions.



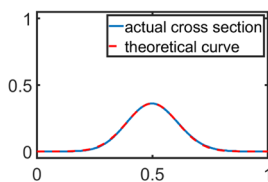
(c) Boundary data at  $y = 0$ .



(d)  $y = \tan(10^\circ) \approx 0.18$ .



(e)  $y = 2 \tan(10^\circ) \approx 0.35$ .



(f)  $y = 3 \tan(10^\circ) \approx 0.53$ .

**Fig. 25** Extrapolating a diagonal line: similarly to Fig. 22, here we consider again the problem of inpainting  $D = [0, 1]^2$  given data on  $[0, 1] \times [-0.3, 0)$ . This time  $u_0$  consists of a line making an angle of  $10^\circ$  with the horizontal, but the slice  $u_0(x, 0)$  is the same step function as in Fig. 22. This time  $D_h$  is  $1000 \times 1000$ px. Inpainting is done using semi-implicit Guidefill ( $r = 3$ ,  $\mathbf{g} = (\cos 10^\circ, \sin 10^\circ)$ ,  $\mu = 100$ ). In c we show the initially sharp signal at  $y = 0$ , while d–f compare horizontal slices at  $y = \tan(10^\circ) \approx 0.18$ ,  $y = 2 \tan(10^\circ) \approx 0.35$  and  $y = 3 \tan(10^\circ) \approx 0.53$  with the predictions of Theorem 2, once again obtaining a very good prediction. Compared with Fig. 22, notice that despite the fact that  $h$  has decreased by an order of magnitude, our loss of signal is much more rapid. This is consistent with the divergence  $\sigma(y, h) \rightarrow \infty$  as  $\theta \rightarrow 0$  we will encounter later in Fig. 23

- Blur gets *better* as  $h \rightarrow 0$ —Fig. 24.
- Blur is *non-existent* if the stencil weights are *degenerate*—that is, put all of their mass into a single *real* pixel  $\mathbf{y}$  (since in this case  $\mathbf{Z}_i$  is deterministic and all variances are 0).
- For semi-implicit Guidefill, blur gets arbitrarily bad as the  $\theta(\mathbf{g}) \rightarrow 0$ .

## 8 Applications to Numerical Linear Algebra

Although we have derived our asymptotic and continuum limits in the context of image inpainting, they are abstract results that apply more generally. In particular, the

asymptotic limit applies to any situation in which a sequence of vectors  $\{\mathbf{x}^{(m)}\}_{m \in \mathbb{N}}$  is generated recursively (for some  $r \in \mathbb{N}$ ) by

$$A_0 \mathbf{x}^{(m)} = A_1 \mathbf{x}^{(m-1)} + A_2 \mathbf{x}^{(m-2)} + \dots + A_r \mathbf{x}^{(m-r)}$$

where  $A_0, A_1, A_2, \dots, A_r$  are circulant or Toeplitz<sup>2</sup> matrices with bandwidth at most  $2r + 1$ , with  $A_0$  an M-matrix (positive diagonal and non-negative off-diagonal entries) while  $A_1, A_2, \dots, A_r$  are element-wise non-negative. An example of a situation where this comes up is the application of damped-Jacobi for iteratively solving the linear system

$$A\mathbf{x} = \mathbf{b}$$

where  $A$  is a Toeplitz or circulant M-matrix. In this case the evolution of the error is given by

$$e^{(m)} = \mathcal{J}_\omega e^{(m-1)} = \mathcal{J}_\omega^m e^{(0)} \quad (67)$$

where  $\omega$  is the damping parameter and the iteration matrix  $\mathcal{J}_\omega$  is given by

$$\mathcal{J}_\omega = I - \omega D^{-1}A,$$

with  $D$  denoting the diagonal of  $A$ . This can be viewed as a special case of the above with  $r$  determined by the bandwidth of  $A$ ,  $A_0 = I$ ,  $A_1 = \mathcal{J}_\omega$ , and

$$A_2 = A_3 = \dots = A_r = O.$$

One may easily verify that  $\mathcal{J}_\omega$  is non-negative so long as  $\omega \in (0, 1)$ . Here we demonstrate how our asymptotic limit may be used to predict the evolution of the error of damped Jacobi applied to the one-dimensional convection diffusion equation

$$u_{xx} + \alpha u_x = f \quad (68)$$

with Dirichlet boundary conditions.

**Boundary Conditions** Technically, our asymptotic limit is only valid for periodic boundary conditions. To make our result rigorous for Dirichlet boundary conditions, we would need to change the stopping time  $\tau$  of the stopped random walk  $\mathbf{X}_\tau := (X_\tau, Y_\tau)$  from Sect. 6.4 to

$$\tau := \inf\{n : X_n < 0 \text{ or } X_n > 1 \text{ or } Y_n < 0\}.$$

However, we know that as  $h \rightarrow 0$  the asymptotic limit approaches the continuum limit, and one may easily see that simply padding the boundary data  $u_0$  on either side

<sup>2</sup> Technically, our result only applies to the circulant case, but we will argue that it applies approximately to the Toeplitz case as well.

with an infinite strip of zeros leads to the same continuum limit—and hence this should also lead to a reasonable approximation for the asymptotic limit, at least away from the boundary at  $x = 0$  and  $x = 1$ .

**Discretization** We consider two discretizations: 2nd order centered differences, leading to

$$A_{\text{centered}} = \text{tri} \left[ -1 - \frac{\alpha h}{2}, 2, -1 + \frac{\alpha h}{2} \right]$$

and first order upwind differences

$$A_{\text{upwind}} = \text{tri} [-1, 2 - \alpha h, -1 + \alpha h]$$

**Centered differences** In this case, we obtain the iteration matrix

$$\mathcal{J}_{\omega, \text{centered}} = \text{tri} \left[ \frac{\omega}{2} \left( 1 + \frac{\alpha h}{2} \right), 1 - \omega, \frac{\omega}{2} \left( 1 - \frac{\alpha h}{2} \right) \right].$$

Assuming  $0 \leq \omega \leq 1$  and  $-2 \leq \alpha h \leq 2$  we have  $\mathcal{J}_{\omega, \text{centered}}$  non-negative, and hence by Sect. 6.4 we have the associated random walk with increments  $(U, V)$  given by

$$U = \begin{cases} 1 & \text{with probability } \frac{\omega}{2} \left( 1 - \frac{\alpha h}{2} \right) \\ 0 & \text{with probability } 1 - \omega \\ -1 & \text{with probability } \frac{\omega}{2} \left( 1 + \frac{\alpha h}{2} \right) \end{cases}$$

and  $V = -1$  with probability 1. Since  $V$  is deterministic, the second form of the asymptotic limit (Theorem 4) applies so long as  $\omega \neq 1$ . If  $\omega = 1$ , the stencil of  $U$  is constrained to the sublattice  $-1 + 2 \cdot \mathbb{Z} \subsetneq \mathbb{Z}$  and hence the conditions of Theorem 4 are violated. We will see in Fig. 26 that in this case the asymptotic limit of Theorem 4 *does not* accurately predict the error evolution (67). To compute the asymptotic limit, we need only consider the mean  $\mu$  and variance  $\sigma^2$  of  $U$ . We have

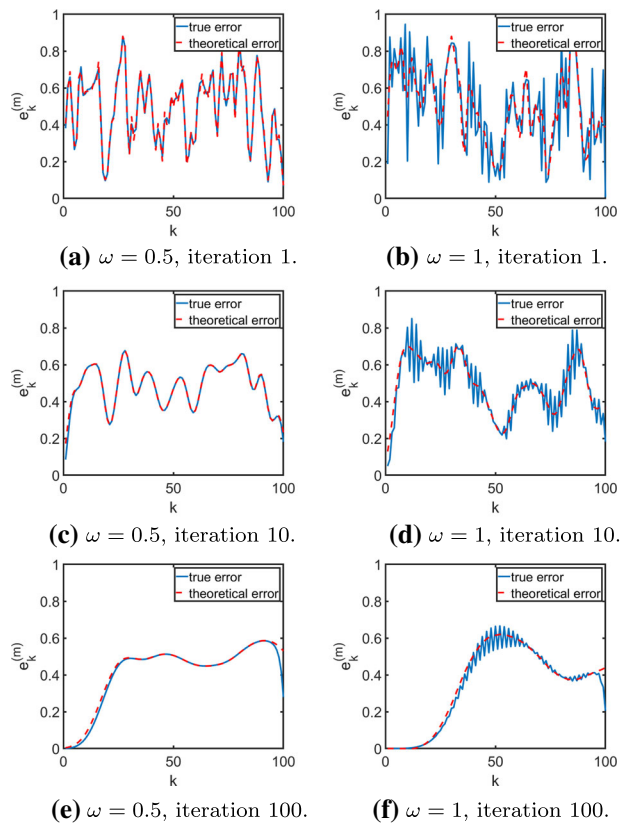
$$\mu = -\omega \frac{\alpha h}{2} \quad \sigma^2 = \omega - \frac{\omega^2 \alpha^2 h^2}{4}$$

Thus, defining

$$g_{\sigma, \mu}(n) := \frac{e^{-\frac{(n-\mu)^2}{2\sigma^2}}}{\sum_{n'=1}^N e^{-\frac{(n'-\mu)^2}{2\sigma^2}}}$$

we have

$$e_k^{(m)} \approx (e^{(0)} * g_{m\sigma, m\mu})_k. \quad (69)$$



**Fig. 26** Error evolution of damped Jacobi: here we plot evolution of the error of damped Jacobi applied to the one dimensional convection diffusion Eq. (68), discretized using centered differences with  $\alpha h = 1$  and on a grid of size  $N = 100$ . Also included is the theoretical error as predicted by the asymptotic limit (Theorem 4). We consider both  $\omega = 1$  and  $\omega = 0.5$ . In the former case, the asymptotic limit captures of the overall trend of the error evolution, but fails to capture some high frequency oscillations. This is because when  $\omega = 1$ , the assumptions of Theorem 4 are not satisfied. On the other hand, when  $\omega = 0.5$ , the conditions of Theorem 4 are satisfied and the asymptotic limit gives an excellent approximation of the true error evolution

Figure 26 presents a comparison of the predicted error (69) with the real measured error, for  $N = 100$ ,  $\alpha h = 1$  fixed and for  $\omega = 0.5$  and  $\omega = 1$ . Note that even in the case  $\omega = 1$ , where the assumptions of Theorem 4 do not apply, (69) is still able to predict the overall *shape* of the error, minus high frequency oscillations. For  $\omega \neq 1$ , (69) is an excellent prediction of the error, except near the boundaries  $x = 0$  and  $x = 1$ .

**Upwind** in this case, we obtain the iteration matrix

$$\mathcal{J}_{\omega, \text{upwind}} = \text{tri} \left[ \frac{\omega}{2 - \alpha h}, 1 - \omega, \omega \frac{1 - \alpha h}{2 - \alpha h} \right].$$

This time we assume  $0 \leq \omega, \alpha h \leq 1$  and consider the associated random walk with increments  $(U, V)$  given by

$$U = \begin{cases} 1 & \text{with probability } \omega \frac{1-\alpha h}{2-\alpha h} \\ 0 & \text{with probability } 1-\omega \\ -1 & \text{with probability } \frac{\omega}{2-\alpha h} \end{cases}$$

and  $V = -1$  with probability 1. Again, the second form of the asymptotic limit (Theorem 4) applies so long as  $\omega \neq 1$ . The mean  $\mu$  and variance  $\sigma^2$  of  $U$  in this case are given by

$$\mu = -\omega \frac{\alpha h}{2-\alpha h} \quad \sigma^2 = \omega - \frac{\omega^2 \alpha^2 h^2}{(2-\alpha h)^2}$$

Once again the predicted error is given by (69). Similar results (omitted for brevity) are obtained as in the centered differences case.

**Remark 16** The *multigrid algorithm* [9,39] is based in part on the observation that simple iterative methods such as damped Jacobi or Gauss–Seidel—while not necessarily effective at *eliminating* the error—can be very effective at *smoothing* it. It is well known that while Jacobi iteration ( $\omega = 1$ ) is typically not an effective smoother, damped Jacobi with  $\omega \neq 1$  often is, and indeed it is widely employed in the multigrid method [39]. The analysis of this section corroborates this well known fact from the multigrid community—for  $\omega \neq 1$ , Theorem 4 gives us an explicit formula for the asymptotic error at iteration  $m$  as “smoothed” version of the initial error. However, when  $\omega = 1$ , the assumptions of Theorem 4 are violated and this fails.

**Remark 17** In its present form, our asymptotic limit cannot be used to analyze the error evolution of damped Jacobi applied to the linear systems arising from the discretization of partial differential equations in dimension  $d \geq 2$ . To handle this case, we must first generalize our asymptotic limit to  $\mathbb{R}^{d+1}$ . This is straightforward. In particular, the original form of Theorem 1 as it is presented in [36], is applicable to random walks in  $\mathbb{R}^d$ . We quoted the two dimensional version of it in Theorem 1 simply because this was the version relevant to the problem at hand. Since the asymptotic limit is derived from Theorem 1, generalizing it to arbitrary dimensions is straightforward.

## 9 Conclusions and Future Work

### 9.1 Conclusions

In this paper we have presented a detailed analysis of a class of geometric inpainting algorithms based on the idea of filling the inpainting domain in successive shells from the boundary inwards, where every pixel to be filled is assigned a color equal to a weighted average of its already filled neighbors. These methods are all special cases of a generic framework sketched in Algorithm 1. Methods in the literature falling within this framework include

- Telea’s Algorithm [38].
- Coherence Transport [7,31].
- Guidefill [27].



A subtle but important point about these methods is that pixels in the current inpainting boundary are filled independently. Noting this, we have proposed a semi-implicit extension of these methods in which pixels in the current shell are instead filled simultaneously by solving a linear system. We have also sketched in Algorithm 1 a straightforward extension of the original framework that is equivalent to solving this linear system using damped Jacobi or successive over-relaxation (SOR). A theoretical convergence analysis of these methods is presented for the semi-implicit extension of Guidefill, where we show that SOR is extremely effective. This analysis is backed up by numerical experiments.

As all of the algorithms listed above (with the exception of semi-implicit Guidefill, which is presented for the first time in this paper) are known to create some disturbing visual artifacts, the main objective of our analysis is to understand why these occur and whether or not they are inevitable. We focus specifically on kinking artifacts and blur artifacts. Other artifacts including the formation of shocks and extrapolated isophotes that end abruptly are discussed but not analyzed, as they have already been studied elsewhere [7,32] and are well understood. Our analysis is based on three main ideas:

- A continuum limit, which we use to explain kinking artifacts.
- An asymptotic limit, which we use to analyze blur artifacts.
- A connection to the theory of stopped random walks, from which both limits are derived.

Similarly to the earlier work of Bornemann and März [7], our continuum limit is a transport equation. However, our limit process is different and so are the coefficients of the resulting transport equation. Moreover, numerical experiments show that our transport equation is a better reflection of the behaviour of Algorithm 1 (the direct form and our proposed semi-implicit extension) in practice, capable of accurately predicting kinking phenomena that is not captured by the alternative continuum limit proposed in [7]. The third core idea of our analysis, which is to relate Algorithm 1 and its extension to stopped random walks, is critical for two reasons. Firstly, it allows us to prove convergence to our continuum limit even for boundary data with low regularity, such as jump discontinuities. By contrast, the analysis in [7] demonstrates convergence under the assumption of smooth boundary data, which is an unrealistic assumption for images. Secondly, this connection is central to our analysis of blur artifacts, which we analyze based not on a continuum limit where  $h \rightarrow 0$ , but rather an asymptotic limit where  $h$  very small but nonzero. Our asymptotic limit allows us to make quantitative predictions regarding blur that are in excellent agreement with numerical experiments, even for relatively low resolution images (e.g. Fig. 22 which is only  $200 \times 200$ px). Although our analysis operates in a highly idealized setting (Sect. 6.1), our conclusions are far reaching. In particular, we prove the following:

1. In the direct form of Algorithm 1, kinking artifacts will always be present. That is, certain isophotes cannot be extended into the inpainting domain without bending (Sect. 7.1.1).
2. This is not true of the semi-implicit extension of Algorithm 1. In particular, semi-implicit Guidefill can extrapolate isophotes with any orientation,<sup>3</sup> and moreover

<sup>3</sup> That is, unless the isophotes are exactly parallel to the boundary of the inpainting domain. But in this case extrapolation is not defined.

is able to do so efficiently by using SOR to solve the required linear system (Sects. 7.1.1, 7.1.2, Corollary 1).

3. Blur artifacts exhibit an angular dependence, which for semi-implicit Guidefill become arbitrarily bad as the angle an extrapolated isophote makes with the boundary of the inpainting domain goes to zero. Thus, semi-implicit Guidefill pays a heavy price (on top of the need to solve a linear system for every shell) for its ability to successfully extrapolate such isophotes (Theorem 2, Fig. 23).
4. Blur artifacts become less significant as the resolution of the image goes up, and get worse the further into the inpainting domain you extrapolate (Theorem 2).
5. Methods that put all of their weight into a single pixel exhibit no blur, but can only extrapolate without kinking in finitely many directions (Sect. 7.3).

In addition to this, we have also demonstrated that our asymptotic limit has applications outside of image processing. In particular, in Sect. 8 we showed how it could be used to predict the error evolution of damped Jacobi applied to a discretized one-dimensional convection-diffusion problem.

## 9.2 Future Work

There are a three main questions we would like to explore in the future.

1. Does there exist an algorithm within the framework we have described that avoids blurring artifacts and kinking artifacts at the same time? If not, is it at least possible to design an algorithm that, like semi-implicit Guidefill, avoids kinking artifacts so long as the guidance direction  $\mathbf{g} = (\cos \theta, \sin \theta)$  obeys  $\theta \neq 0$ , but for which the severity of blur as a function of  $\theta$  remains bounded? Could this be done, for example, by the replacing the bilinear interpolation used to define ghost pixels with a more sophisticated interpolation scheme?
2. What happens if the semi-implicit version of Algorithm 1 is generalized to a fully-implicit extension in which pixel colors are computed as a weighted average not only of their (known) already filled neighbors in  $A_{\epsilon,h}(\mathbf{x}) \cap \Omega \setminus D^{(k)}$  and (unknown) neighbors within the same shell  $\partial D_h^{(k)}$ , but of *all* neighbors in  $A_{\epsilon,h}(\mathbf{x})$ ? Are their additional benefits in terms of artifact reduction and, if so, can the resulting linear system be solved efficiently enough to make this worthwhile?
3. We would like to explore the connection of our asymptotic limit to problems in numerical linear algebra more deeply. In particular, we would like to generalize our asymptotic limit to arbitrary dimensions so that it can be applied to discretized partial differential equations in dimension  $d \geq 2$  (we have already sketched how to do this in Remark 17). We would also like to see whether our results can be applied to more sophisticated methods than damped Jacobi and—if so—whether or not any new insights can be gained by doing so.

**Acknowledgements** The authors would like to thank Vittoria Silvestri, James Norris, and Arie Iserles for helpful conversations. Thanks is also due to Dömötör Pálvölgyi, whose idea of a symmetry-based argument eventually led to “Appendix E”, Proposition 5. Finally, we would like to thank an anonymous referee for their insightful comments and suggestions.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## A Punctured Sums

Here we provide further justification for our exclusion of the point  $\mathbf{x}$  from the update formula (1) in Algorithm 1 as mentioned in Remark 4. As we mentioned there, this makes no difference to the direct form of Algorithm 1, because the subroutine FillRow only involves sums taken over  $A_{\epsilon,h}(\mathbf{x}) \cap (\Omega \setminus D^{(k)})$ , which never contains  $\mathbf{x}$ . However, the semi-implicit extension of Algorithm 1 expresses  $u_h(\mathbf{x})$  as a sum of  $u_h(\mathbf{y})$  over a set of points that might include  $\mathbf{x}$ . The reason we deliberately exclude  $\mathbf{x}$  is because, as the following proposition shows, if  $w_\epsilon(\mathbf{x}, \mathbf{x}) < \infty$ , it doesn't matter, but if  $w_\epsilon(\mathbf{x}, \mathbf{x}) = \infty$ , it wreaks havoc. Moreover, the weights (6) which we wish to study *do* have the property that  $w_\epsilon(\mathbf{x}, \mathbf{x}) = \infty$ .

**Proposition 2** *Suppose  $\mathbf{x} \in B$  for some finite set  $B \subset \mathbb{R}^2$ , and suppose there exist non negative weights  $w : B \times B \rightarrow [0, \infty]$ , finite everywhere except possibly at  $(\mathbf{x}, \mathbf{x})$ . Then if  $w(\mathbf{x}, \mathbf{x}) < \infty$ , we have*

$$u_h(\mathbf{x}) = \frac{\sum_{\mathbf{y} \in B} w(\mathbf{x}, \mathbf{y}) u_h(\mathbf{y})}{\sum_{\mathbf{y} \in B} w(\mathbf{x}, \mathbf{y})} = \frac{\sum_{\mathbf{y} \in B \setminus \{\mathbf{x}\}} w(\mathbf{x}, \mathbf{y}) u_h(\mathbf{y})}{\sum_{\mathbf{y} \in B \setminus \{\mathbf{x}\}} w(\mathbf{x}, \mathbf{y})}.$$

On the other hand, if  $w(\mathbf{x}, \mathbf{x}) = \infty$ , we have an indeterminate expression

$$u_h(\mathbf{x}) = \frac{\infty}{\infty}.$$

**Proof** The proof is an exercise in cancellation and left to the reader.  $\square$

## B Properties of Ghost Pixels and Equivalent Weights

In this appendix we prove the six properties of ghost pixels and equivalent weights listed in Sect. 4. These properties all follow from the definition of  $u_h(\mathbf{y})$  when  $\mathbf{y}$  is a ghost pixel, namely

$$u_h(\mathbf{y}) = \sum_{\mathbf{z} \in \text{Supp}(\{\mathbf{y}\})} \Lambda_{\mathbf{z},h}(\mathbf{y}) u_h(\mathbf{z}),$$

where  $\{\Lambda_{\mathbf{z},h}\}_{\mathbf{z} \in \mathbb{Z}_h^2}$  denote the basis functions of bilinear interpolation associated with the lattice  $\mathbb{Z}_h^2$ , and where  $\text{Supp}(A)$  denotes the set of real pixels needed to define a set

A of ghost pixels using bilinear interpolation. Note that if  $\mathbf{y} \in A$ , then

$$\Lambda_{\mathbf{z},h}(\mathbf{y}) = 0 \quad \text{for all } \mathbf{z} \notin \text{Supp}(A). \quad (70)$$

The following explicit formula for  $\text{Supp}(\{(x, y)\})$  (which comes from the definition of bilinear interpolation) will occasionally be useful.

$$\text{Supp}(\{(x, y)\}) = \left\{ \left( \left\lfloor \frac{x}{h} \right\rfloor h, \left\lfloor \frac{y}{h} \right\rfloor h \right), \left( \left\lceil \frac{x}{h} \right\rceil h, \left\lfloor \frac{y}{h} \right\rfloor h \right), \right. \\ \left. \left( \left\lfloor \frac{x}{h} \right\rfloor h, \left\lceil \frac{y}{h} \right\rceil h \right), \left( \left\lceil \frac{x}{h} \right\rceil h, \left\lceil \frac{y}{h} \right\rceil h \right) \right\}. \quad (71)$$

First we prove property one.

1. *Explicit formula:*

$$\tilde{w}(\mathbf{x}, \mathbf{z}) = \sum_{\mathbf{y} \in A(\mathbf{x})} \Lambda_{\mathbf{y},h}(\mathbf{z}) w(\mathbf{x}, \mathbf{y})$$

**Proof** This follows straightforwardly from the definition of ghost pixels, the property (70), and a few exchanges of finite sums.

$$\begin{aligned} u_h(\mathbf{x}) &= \sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) u_h(\mathbf{y}) \\ &= \sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) \sum_{\mathbf{z} \in \text{Supp}(\{\mathbf{y}\})} \Lambda_{\mathbf{z},h}(\mathbf{y}) u_h(\mathbf{z}) \\ &= \sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) \sum_{\mathbf{z} \in \text{Supp}(A(\mathbf{x}))} \Lambda_{\mathbf{z},h}(\mathbf{y}) u_h(\mathbf{z}) \\ &= \sum_{\mathbf{z} \in \text{Supp}(A(\mathbf{x}))} \underbrace{\left\{ \sum_{\mathbf{y} \in A(\mathbf{x})} \Lambda_{\mathbf{z},h}(\mathbf{y}) w(\mathbf{x}, \mathbf{y}) \right\}}_{:= \tilde{w}(\mathbf{x}, \mathbf{z})} u_h(\mathbf{z}) \end{aligned}$$

□

Next, instead of proving properties two and three, we prove a stronger result from which they both follow.

1.(a) *Preservation of degree 1 polynomials* Suppose  $f(\mathbf{y})$  is a (scalar valued) polynomial of degree at most 1, that is  $f(\mathbf{y}) = a + \mathbf{b} \cdot \mathbf{y}$  for some  $a \in \mathbb{R}$  and  $\mathbf{b} \in \mathbb{R}^2$ . Then

$$\sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) f(\mathbf{y}) = \sum_{\mathbf{y} \in \text{Supp}(A(\mathbf{x}))} \tilde{w}(\mathbf{x}, \mathbf{y}) f(\mathbf{y}).$$

**Proof** This follows from the fact that the bilinear interpolant of a polynomial of degree at most 1 is just the polynomial back. That is,

$$\sum_{\mathbf{z} \in \text{Supp}(\{\mathbf{y}\})} \Lambda_{\mathbf{z},h}(\mathbf{y}) f(\mathbf{z}) = f(\mathbf{y}).$$

This is obvious and we do not prove it. We will also use (70) once.

$$\begin{aligned} & \sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) f(\mathbf{y}) \\ &= \sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) \left\{ \sum_{\mathbf{z} \in \text{Supp}(\{\mathbf{y}\})} \Lambda_{\mathbf{z},h}(\mathbf{y}) f(\mathbf{z}) \right\} \\ &= \sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) \left\{ \sum_{\mathbf{z} \in \text{Supp}(A(\mathbf{x}))} \Lambda_{\mathbf{z},h}(\mathbf{y}) f(\mathbf{z}) \right\} \\ &= \sum_{\mathbf{z} \in \text{Supp}(A(\mathbf{x}))} \left\{ \sum_{\mathbf{y} \in A(\mathbf{x})} \Lambda_{\mathbf{z},h}(\mathbf{y}) w(\mathbf{x}, \mathbf{y}) \right\} f(\mathbf{z}) \\ &= \sum_{\mathbf{z} \in \text{Supp}(A(\mathbf{x}))} \tilde{w}(\mathbf{x}, \mathbf{z}) f(\mathbf{z}). \end{aligned}$$

□

## 2. Preservation of total mass.

$$\sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{y} \in \text{Supp}(A(\mathbf{x}))} \tilde{w}(\mathbf{x}, \mathbf{y}).$$

**Proof** Special case of 1.(a),  $p(\mathbf{y}) \equiv 1$ .

□

## 3. Preservation of center of mass (or first moment).

$$\sum_{\mathbf{y} \in A(\mathbf{x})} w(\mathbf{x}, \mathbf{y}) \mathbf{y} = \sum_{\mathbf{y} \in \text{Supp}(A(\mathbf{x}))} \tilde{w}(\mathbf{x}, \mathbf{y}) \mathbf{y}.$$

**Proof** Apply 1.(a) to each component of  $f(\mathbf{y}) = \mathbf{y}$ .

□

## 4. Inheritance of non-negativity:

$$\tilde{w}_\epsilon(\mathbf{x}, \mathbf{z}) \geq 0 \quad \text{for all } \mathbf{z} \in \text{Supp}(A_{\epsilon,h}(\mathbf{x})).$$

**Proof** This is immediate from the non-negativity of the original weights  $\{w_\epsilon\}$ , the non-negativity of the basis functions  $\{\Lambda_{\mathbf{y},h}\}$ , and the explicit formula (9). □

### 5. Inheritance of non-degeneracy condition (3).

$$\sum_{\mathbf{y} \in \text{Supp}(A_{\epsilon,h}(\mathbf{x}) \cap (\Omega \setminus D^{(k)}))} \tilde{w}_{\epsilon}(\mathbf{x}, \mathbf{y}) > 0.$$

**Proof** Apply preservation of total mass to (3). □

6. *Universal Support.* For any  $n \in \mathbb{Z}$ , we have

$$\begin{aligned} \text{Supp}(A_{\epsilon,h}(\mathbf{x}) \cap \{y \leq nh\}) &\subseteq D_h(B_{\epsilon,h}(\mathbf{x})) \cap \{y \leq nh\} \\ &\subseteq B_{\epsilon+2h,h}(\mathbf{x}) \cap \{y \leq nh\}. \end{aligned}$$

where  $\{y \leq nh\} := \{(x, y) \in \mathbb{R}^2 : y \leq nh\}$ , and where  $D_h$  is the dilation operator defined in our section on notation.

**Proof** Let  $(x, y) \in A_{\epsilon,h}(\mathbf{x}) \cap \{y \leq nh\}$ . Then  $x^2 + y^2 \leq \epsilon^2$  and  $y \leq nh$ . Hence  $(\lfloor \frac{x}{h} \rfloor h, \lfloor \frac{y}{h} \rfloor h) \in B_{\epsilon,h}(\mathbf{x})$ , and by (71) we have

$$\text{Supp}(\{(x, y)\}) \subseteq \mathcal{N}\left(\left(\left\lfloor \frac{x}{h} \right\rfloor h, \left\lfloor \frac{y}{h} \right\rfloor h\right)\right) \subseteq D_h(B_{\epsilon,h}(\mathbf{x})),$$

where  $\mathcal{N}(\mathbf{x})$  denotes the nine-point neighborhood of  $\mathbf{x} \in \mathbb{Z}_h^2$  defined in the notation section. But since we also know  $y \leq nh$ , we have  $\lfloor \frac{y}{h} \rfloor h \leq \lfloor \frac{nh}{h} \rfloor h \leq nh$ , and hence applying (71) again we conclude

$$\text{Supp}(\{(x, y)\}) \subseteq \{y \leq nh\}$$

as well. Since  $(x, y) \in A_{\epsilon,h}(\mathbf{x}) \cap \{y \leq nh\}$  was arbitrary, the first inclusion follows. For the second inclusion, note that every element of  $D_h(B_{\epsilon,h}(\mathbf{x})) \cap \{y \leq nh\}$  is of the form  $(x, y) = \mathbf{y} + \Delta \mathbf{y}$  where  $\mathbf{y} \in B_{\epsilon,h}(\mathbf{x})$  and  $\Delta \mathbf{y} \in \{-h, 0, h\} \times \{-h, 0, h\}$ . Hence

$$\|(x, y)\| = \|\mathbf{y} + \Delta \mathbf{y}\| \leq \|\mathbf{y}\| + \|\Delta \mathbf{y}\| \leq \epsilon + \sqrt{2}h < \epsilon + 2h.$$

At the same time we have  $y \leq nh$ , so  $(x, y) \in B_{\epsilon+2h,h}(\mathbf{x}) \cap \{y \leq nh\}$  as claimed. □

## C Proof that Our Proposed Extension of Algorithm 1 is Equivalent to Damped Jacobi or SOR

Here we supply the proof, promised in Sect. 5, that the blue text in Algorithm 1 is actually an implementation of either damped Jacobi or SOR, depending on whether the “FillBoundary” subroutine is executed sequentially or in parallel.

**Proposition 3** *Changing the boolean “semiImplicit” to true in Algorithm 1 is equivalent to solving (15) with damped Jacobi if “FillBoundary” is executed in parallel,*

and with SOR if it is executed sequentially. In either case, the relaxation parameter is given by

$$\omega^* = \left(1 - \frac{\tilde{w}_\epsilon(\mathbf{x}, \mathbf{x})}{W}\right).$$

**Proof** First, note that the Jacobi iteration for solving the linear system (15) may be written as

$$\tilde{u}_h^{(n+1)}(\mathbf{x}) = \frac{1}{1 - \frac{\tilde{w}_\epsilon(\mathbf{x}, \mathbf{x})}{W}} \left( \sum_{\mathbf{y} \in S_{\epsilon, h}^{(k)}(\mathbf{x}) \setminus \{\mathbf{x}\}} \frac{\tilde{w}_\epsilon(\mathbf{x}, \mathbf{y})}{W} u_h^{(n)}(\mathbf{y}) + \mathbf{f} \right)$$

with  $\mathbf{f}$  defined as in (18). By comparison, repeated (parallel) execution of  $\text{FillBoundary}(D_h^{(k+1)}, \partial D_h^{(k)})$  is equivalent (after applying the definition of equivalent weights) to

$$\begin{aligned} u_h^{(n+1)}(\mathbf{x}) &= \sum_{\mathbf{y} \in S_{\epsilon, h}^{(k)}(\mathbf{x})} \frac{\tilde{w}_\epsilon(\mathbf{x}, \mathbf{y})}{W} u_h^{(n)}(\mathbf{y}) + \mathbf{f} \\ &= \frac{\tilde{w}(\mathbf{x}, \mathbf{x})}{W} u_h^{(n)}(\mathbf{x}) \\ &\quad + \sum_{\mathbf{y} \in S_{\epsilon, h}^{(k)}(\mathbf{x}) \setminus \{\mathbf{x}\}} \frac{\tilde{w}_\epsilon(\mathbf{x}, \mathbf{y})}{W} u_h^{(n)}(\mathbf{y}) + \mathbf{f} \\ &= (1 - \omega^*) u_h^{(n)}(\mathbf{x}) + \omega^* \tilde{u}_h^{(n+1)}(\mathbf{x}), \end{aligned}$$

which is a definition of damped Jacobi. The proof for SOR is analogous.  $\square$

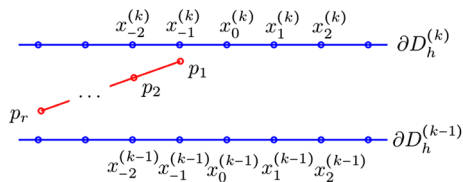
## D Proof of Proposition 1

First we fix some notation. Suppose we are on iteration  $k$  of semi-implicit GuidedFill and let  $\mathbf{x} := x_0^{(k)}$  denote a fixed but arbitrary member of  $\partial D_h^{(k)}$ . The pixel  $x_0^{(k)}$  is coupled by (15) to its immediate neighbors  $x_j^{(k)}$  for  $-r - 2 \leq j \leq r + 2$ , and also depends on the pixels  $x_j^{(k-\delta)} := \mathbf{x} + h(j, \delta) \in \partial D_h^{(k-\delta)}$  for  $(j, \delta) \in b_{r+2}^-$  which appear in the right hand side of (15) within the vector  $\mathbf{f}$ .

Next, note that since  $\mu \rightarrow \infty$ , the weights  $w_r$  have vanished on all of  $b_r^0$  except for the line of ghost pixels

$$\ell_r^- := \{-j\mathbf{g}\}_{j=1}^r.$$

For convenience, we enumerate  $\ell_r^-$  as  $\ell_r^- := \{p_j\}_{j=1}^r$  where  $p_j = -j\mathbf{g}$ . Each  $p_j$  receives weight



**Fig. 27** Illustration of the position of the line  $\ell_r^-$  relative to the current shell  $\partial D_h^{(k)}$  and previous shell  $\partial D_h^{(k-1)}$ : here we visualize the line  $\ell_r^- := \{-j\mathbf{g}\}_{j=1}^r \subseteq \tilde{b}_r^-$  when  $\mathbf{g} = (\cos \theta, \sin \theta)$  with  $0 < \theta < \theta_c = \arcsin(1/r)$ . In this case  $\ell_r^-$  fits entirely into the space between  $\partial D_h^{(k)}$  and  $\partial D_h^{(k-1)}$ . For convenience, we enumerate  $\ell_r^-$  as  $\ell_r^- := \{p_j\}_{j=1}^r$  where  $p_j = -j\mathbf{g}$ . We write the current pixel of interest  $\mathbf{x}$  as  $x_0^{(k)}$  for convenience, and its unknown neighbors in  $\partial D_h^{(k)}$  as  $x_j^{(k)}$  for  $-r-2 \leq j \leq r+2$ , while its already filled neighbors (we only show the ones in  $\partial D_h^{(k-1)}$ ) are denoted by  $x_j^{(k-\delta)} := \mathbf{x} + h(j, \delta) \in \partial D_h^{(k-\delta)}$  for  $(j, \delta) \in b_{r+2}^-$

$$w_j := \frac{1}{j}.$$

This situation is illustrated in Fig. 27 for the case  $0 < \theta < \theta_c$ , where  $\ell_r^-$  fits entirely into the space between  $\partial D_h^{(k)}$  and  $\partial D_h^{(k-1)}$ .

To compute the entries of  $\mathcal{L}$ , we follow the idea of Sect. 4 and consider how the weight  $w_j$  of each ghost pixel  $p_j$  gets distributed to its real pixel neighbors. For example, in Fig. 27, the weight  $w_1$  of  $p_1$  gets redistributed amongst the four pixels  $x_0^{(k)}$ ,  $x_{-1}^{(k)}$ ,  $x_0^{(k-1)}$ , and  $x_{-1}^{(k-1)}$ .

How exactly this weight gets redistributed is for the most part not something we need to know precisely. For example, it is already clear from Fig. 27 that if  $0 < \theta \leq \frac{\pi}{2}$ , then none of the weight of any of the  $p_j$  make it into any of  $x_1^{(k)}$ ,  $x_2^{(k)}$ ,  $x_3^{(k)}$  ... Similarly, if  $\frac{\pi}{2} \leq \theta < \pi$ , no weight makes it to any of  $x_{-1}^{(k)}$ ,  $x_{-2}^{(k)}$ ,  $x_{-3}^{(k)}$  ... This means that, given our assumed ordering of pixels within each layer  $\partial D_h^{(k)}$ , we already know that  $\mathcal{L}$  is a lower triangular matrix. Hence  $L = \mathcal{L}$ ,  $U = O$ . Therefore,  $G_\omega$  takes on the simplified form

$$G_\omega = (1 - \omega)(I - \omega D^{-1}L)^{-1}.$$

We begin with  $\|G_\omega\|_\infty$ , the harder case. In this case, defining

$$A := I - \omega D^{-1}L,$$

we have

$$\|G_\omega\|_\infty = \|1 - \omega\| \|A^{-1}\|_\infty \quad (72)$$

We know  $\mathcal{L} = D - L$  is strictly diagonally dominant, so the following computation shows that  $A$  is as well, provided  $0 < \omega \leq 1$ :



$$\sum_{j \neq i} |A_{ij}| = \frac{|\omega|}{|D_{ii}|} \sum_{j \neq i} |L_{ij}| < |\omega| \leq 1 = |A_{ii}|.$$

Hence, the following classical bound due to Jim Varah [41, Theorem 1] applies:

$$\|A^{-1}\|_{\infty} \leq \frac{1}{\min_{i=1}^N \Delta_i(A)}$$

where

$$\Delta_i(A) := |A_{ii}| - \sum_{j \neq i} |A_{ij}|.$$

Since  $A$  is a Toeplitz matrix with band width  $r + 2$  and at the same time a lower triangular matrix, we know that  $\Delta_i(A)$  is the same for all  $i \geq r + 3$ , but increases somewhat for  $i \leq r + 2$  as there are fewer off diagonal terms (due to our assumed Dirichlet boundary conditions). In particular, the first row has *no* off diagonal terms, so we have  $\Delta_1(A) = A_{11} = 1$ . It follows that

$$\Delta_1(A) \geq \Delta_2(A) \geq \dots \geq \Delta_{r+3}(A) = \Delta_{r+4}(A) = \dots = \Delta_N(A).$$

Choosing row  $N$  as a representative row for convenience gives

$$\|A^{-1}\| \leq \frac{1}{\Delta_N(A)}.$$

However, the identity

$$\|A^{-1}\|^{-1} = \inf_{\mathbf{x}} \frac{\|A\mathbf{x}\|}{\|\mathbf{x}\|}$$

(valid for any induced norm) means that in particular, for the vector  $e$  containing  $r + 2$  zeros followed by  $N - r - 2$  ones, that is

$$e = (\underbrace{0, \dots, 0}_{r+2}, \underbrace{1, \dots, 1}_{N-r-2}),$$

we have

$$\begin{aligned} \|A^{-1}\|_{\infty}^{-1} &\leq \frac{\|Ae\|_{\infty}}{\|e\|_{\infty}} = \max_{i=1}^N \left| \sum_{j=1}^N A_{ij} e_j \right| \\ &= \max_{i=r+3}^N \left| A_{ii} - \sum_{j \neq i} |A_{ij}| \right| = \Delta_N(A). \end{aligned}$$

where we have used the fact that for all  $i$  we have  $A_{ii} > 0$  and  $A_{ij} \leq 0$  for  $j \neq i$ . The vector  $e$  was chosen deliberately in order to avoid the first  $r + 2$  rows of  $A$ , which we have already said are different due to boundary conditions. Hence

$$\|A^{-1}\|_{\infty} \geq \frac{1}{\Delta_N(A)}$$

as well, and having proven the bound in both directions we conclude

$$\|A^{-1}\|_{\infty} = \frac{1}{\Delta_N(A)}. \quad (73)$$

**Remark 18** It appears that Varah's bound [41, Theorem 1] should generalize to equality not only in our case, but to general strictly diagonally dominant Toeplitz matrices obeying  $A_{ii} > 0$  for all  $i$  and  $A_{ij} \leq 0$  whenever  $j \neq i$ , using a very similar argument. However, this generalization does not appear in [41] and we have been unable to find it in the literature.

The next step is to compute  $\Delta_N(A)$ . To that end, note that by definition  $A = I - \omega D^{-1}L$  obeys  $A_{ii} = 1$  for all  $i$  and

$$A_{ij} = -\omega \frac{\left( \frac{\tilde{w}_r(\mathbf{0}, (j-i)\mathbf{e}_1)}{W} \right)}{\left( 1 - \frac{\tilde{w}_r(\mathbf{0}, \mathbf{0})}{W} \right)} = -\omega \frac{\tilde{w}_r(\mathbf{0}, (j-i)\mathbf{e}_1)}{W - \tilde{w}_r(\mathbf{0}, \mathbf{0})}$$

for

$$\max(i - r - 2, 1) \leq j < i.$$

by (27). Here  $W$  are the total weight and equivalent weights  $\tilde{w}_r$  defined in Sect. 6.1. Hence

$$\Delta_N(A) = \left| 1 - \omega \frac{\sum_{j=-r-2}^{-1} \tilde{w}_r(\mathbf{0}, j\mathbf{e}_1)}{W - \tilde{w}_r(\mathbf{0}, \mathbf{0})} \right| = \left| 1 - \omega \left( \frac{\tilde{W} - \tilde{w}_{0,0}}{W - \tilde{w}_{0,0}} \right) \right|,$$

where  $\tilde{W}$  and  $\tilde{w}_{0,0}$  are defined as in (27). Combining the above with (72) and (73) we finally obtain

$$\|G_{\omega}\|_{\infty} = \frac{|1 - \omega|}{1 - \omega \left( \frac{\tilde{W} - \tilde{w}_{0,0}}{W - \tilde{w}_{0,0}} \right)}$$

as claimed. We leave deriving expressions for  $W$ ,  $\tilde{W}$ , and  $\tilde{w}_{0,0}$  until the end. First we derive an expression for  $\|J_{\omega}\|_{\infty}$  in terms of these three quantities. Since  $U = O$  we have

$$J_{\omega} = I - \omega D^{-1}L = I - \omega D^{-1}(D - L) = (1 - \omega)I + \omega D^{-1}L.$$

By definition we have

$$\|J_\omega\|_\infty := \max_{i=1}^N \sum_{j=1}^N |(J_\omega)_{ij}|.$$

So long as  $i \geq r + 3$ , this sum becomes

$$\sum_{j=1}^N |(J_\omega)_{ij}| = |1 - \omega| + \omega \left( \frac{\tilde{W} - \tilde{w}_{0,0}}{W - \tilde{w}_{0,0}} \right).$$

If  $i \leq r + 2$ , then this sum includes fewer terms and is potentially smaller. Hence

$$\|J_\omega\|_\infty = |1 - \omega| + \omega \left( \frac{\tilde{W} - \tilde{w}_{0,0}}{W - \tilde{w}_{0,0}} \right).$$

Our remaining task is to derive the claimed expressions for  $W$ ,  $\tilde{W}$ , and  $\tilde{w}_{0,0}$ . The easiest is  $W$ . By (17) we have

$$W = \sum_{j=1}^r w_j = \sum_{j=1}^r \frac{1}{j}.$$

It is also not difficult to compute  $\tilde{w}_{0,0}$ , which represents fraction of the mass  $w_1 = 1$  of the point  $p_1$  that gets redistributed back to  $x_0^{(k)}$  (see Fig. 27). Since  $p_1$  sits  $h \sin \theta$  units below  $\partial D_h^{(k)}$  and  $h(1 - \sin \theta)$  units above  $\partial D_h^{(k-1)}$ , and either  $h \cos \theta$  units to the left  $x_0^{(k)}$  and  $h(1 - \cos \theta)$  units right of  $x_{-1}^{(k)}$  if  $0 \leq \theta \leq \frac{\pi}{2}$  or  $h|\cos \theta|$  units right of  $x_0^{(k)}$  and  $h(1 - |\cos \theta|)$  units left of  $x_1^{(k)}$  otherwise, it follows that

$$\tilde{w}_{0,0} = (1 - \sin \theta)(1 - |\cos \theta|)w_1 = (1 - \sin \theta)(1 - |\cos \theta|).$$

For  $\tilde{W}$ , we split into cases. If  $0 \leq \theta \leq \theta_c$  or  $\pi - \theta_c \leq \theta \leq \pi$ , then  $\ell_r^-$  fits entirely between  $\partial D_h^{(k)}$  and  $\partial D_h^{(k-1)}$ , as in Fig. 27. If  $\theta_c < \theta < \pi - \theta_c$ , then only  $p_1$  up to  $p_{j^*}$  fit (recall the definition of  $j^*$  from the statement of the proposition). As a result, in the first case every  $p_j$  for  $1 \leq j \leq r$  contribute mass to  $\tilde{W}$ . In the second case, only the first  $j^*$  contribute. Each contributing  $p_j$  is situated  $hj \sin \theta$  units below  $\partial D_h^{(k)}$  and  $h(1 - j \sin \theta)$  units above  $\partial D_h^{(k-1)}$ . Hence each contributing  $p_j$  contributes  $(1 - j \sin \theta)w_j$  towards  $\tilde{W}$ . Hence, in the first case we have

$$\tilde{W} = \sum_{j=1}^r (1 - j \sin \theta) \frac{1}{j} = \sum_{j=1}^r \frac{1}{j} - r \sin \theta,$$

while in the second we have

$$\tilde{W} = \sum_{j=1}^{j^*} \frac{1}{j} - j^* \sin \theta.$$

Our final claim on the expressions for  $\|J_1\|_\infty$  and  $\|G_1\|_\infty$  and the optimality of  $\omega = 1$  is now a simple exercise and is left to the reader.  $\square$

## E Additional Details on Coherence Transport and the Angular Spectrum

In Sect. 7.1.2 we related the limiting transport direction  $\mathbf{g}_r^* = \lim_{\mu \rightarrow \infty} \mathbf{g}_{\mu,r}^*$  of coherence transport to the angular spectrum  $\Theta(b_r^-)$  of  $b_r^-$  defined by (62). More precisely, first we connected  $\mathbf{g}_r^*$  to the set of minimizers  $\Psi$  within  $b_r^-$  of the orthogonal distance to the line  $L_{\mathbf{g}} = \{\lambda \mathbf{g} : \lambda \in \mathbb{R}\}$ , where  $\mathbf{g}$  is the guidance direction of coherence transport. Then we claimed that  $\Psi$  and  $\Theta(b_r^-)$  are related. Now is the time to prove that claim. We will do this in Proposition 4 not just for  $b_r^-$ , but for a general finite subset  $A \subseteq \mathbb{Z}^2 \cap \{y \leq -1\}$ . To do this, however, first we generalize the concept of angular spectrum to a general subset  $A \subseteq \mathbb{Z}^2$ .

**Definition 8** Given  $A \subseteq \mathbb{Z}^2$  we define the angular spectrum of  $A$  by

$$\Theta(A) = \{\theta \in [0, \pi) : \theta = \theta(\mathbf{y}) \text{ for some } \mathbf{y} \in A \setminus \{\mathbf{0}\}\} \quad (74)$$

If  $A$  is finite it follows that  $\Theta(A)$  is as well, and we write

$$\Theta(A) = \{0 \leq \theta_1 < \theta_2 < \dots < \theta_n < \pi\}.$$

See Fig. 28b for an illustration of  $\Theta(A)$  in the case  $A = b_r^-$ .

Once again we have defined  $\Theta(A)$  modulo  $\pi$  to reflect the fact that  $\mathbf{g}_r^*$  and  $-\mathbf{g}_r^*$  define the same transport equation. The characterization of  $\Theta(A)$  is of interest in and of itself and has been studied for  $A = b_r$  by the likes of Erdős [19] and many others, see for example [15] (they, however, do not define it modulo  $\pi$ ).

**Remark 19** The point of generalizing the concept of angular spectrum and generalizing Proposition 4 from  $b_r^-$  to a general  $A \subseteq \mathbb{Z}^2 \cap \{y \leq -1\}$  is so that we can show (Corollary 2) that our kinking results for coherence transport from Sect. 7.1.2 continue to hold, essentially unchanged, if coherence transport is modified to sum over a discrete square, for example, rather than a discrete ball.

**Proposition 4** Let  $A \subseteq \mathbb{Z}^2 \cap \{y \leq -1\}$  obey  $|A| < \infty$ , and let  $\Theta(A) = \{\theta_1, \theta_2, \dots, \theta_n\}$  denote the angular spectrum of  $A$ , and assume  $n = |\Theta(A)| \geq 2$  in order to avoid degenerate cases (that is, the elements of  $A$  do not all sit on a single line through the origin). Let  $\mathbf{g}_\theta = (\cos \theta, \sin \theta)$  and denote by  $\Psi_\theta$  the set of minimizers of  $|\mathbf{g}_\theta^\perp \cdot \mathbf{y}|$  over

$\mathbf{y} \in A$  (that is, the point(s) in  $A$  minimizing the orthogonal distance to the line  $L_{\mathbf{g}_\theta}$ ). Given  $\mathbf{y} \in A$ , we say that  $\mathbf{y}$  is a singleton minimizer if there is some  $\theta \in [0, \pi)$  for which  $\Psi_\theta = \{\mathbf{y}\}$ . Let  $Y := \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{n'}\}$  denote the set of all singleton minimizers, ordered so that  $\theta(\mathbf{y}_i) \leq \theta(\mathbf{y}_{i+1})$  for all  $i = 1, \dots, n' - 1$ . Then  $n' = n$ ,

$$\Theta(A) = \{\theta(\mathbf{y}_1), \theta(\mathbf{y}_2), \dots, \theta(\mathbf{y}_n)\},$$

and moreover  $\theta_i = \theta(\mathbf{y}_i)$  for all  $i = 1, \dots, n$ . Finally, each singleton minimizer  $\mathbf{y}_i$  is the shortest vector in  $A$  such that  $\theta(\mathbf{y}) = \theta_i$ , that is for every  $\mathbf{y} \in A$  such that  $\theta(\mathbf{y}) = \theta_i$ , we have  $\|\mathbf{y}\| \geq \|\mathbf{y}_i\|$ .

**Proof** Let  $\theta_i \in \Theta(A)$ . Our main objective is to show that  $\theta_i = \theta(\mathbf{y}_i)$ . To achieve that, it suffices to show that the sets  $\Theta(A)$  and  $\{\theta(\mathbf{y}_1), \theta(\mathbf{y}_2), \dots, \theta(\mathbf{y}_{n'})\}$  are equal, since from here it follows that  $n' = n$ , and then the desired identity follows from the ordering property  $\theta(\mathbf{y}_i) \leq \theta(\mathbf{y}_{i+1})$  for all  $i = 1, \dots, n - 1$ . Our secondary objective, to show that  $\mathbf{y}_i$  is the shortest vector in  $A$  obeying  $\theta(\mathbf{y}) = \theta_i$  is something that will be proved along the way.

For the first step, the inclusion

$$\Theta(A) \supseteq \{\theta(\mathbf{y}_1), \theta(\mathbf{y}_2), \dots, \theta(\mathbf{y}_{n'})\}$$

is obvious and follows from the definition of  $\Theta(A)$ . Hence it suffices to prove

$$\Theta(A) \subseteq \{\theta(\mathbf{y}_1), \theta(\mathbf{y}_2), \dots, \theta(\mathbf{y}_{n'})\}. \quad (75)$$

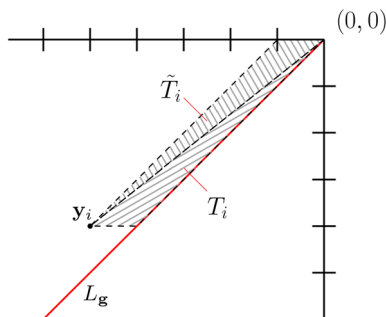
Still fixing  $\theta_i \in \Theta(A)$ , by definition we have  $\theta_i = \theta(\mathbf{y})$  for some  $\mathbf{y} \in A$ . In fact, we have  $\theta_i = \theta(\mathbf{y})$  for all  $\mathbf{y} \in \Psi_{\theta_i}$ , which in this case is a set of vectors that are all scalar multiples of  $\mathbf{g}_{\theta_i}$  and hence all of which obey  $|\mathbf{g}_{\theta_i}^\perp \cdot \mathbf{y}| = 0$ . Define the functions  $\Delta(\theta)$  and  $\delta(\theta)$  by

$$\begin{aligned} \delta(\theta) &:= \max_{\mathbf{y} \in \Psi_\theta} |\mathbf{g}_\theta^\perp \cdot \mathbf{y}| \\ \Delta(\theta) &:= \begin{cases} \min_{\mathbf{y} \in A \setminus \Psi_\theta} |\mathbf{g}_\theta^\perp \cdot \mathbf{y}| & \text{if } A \setminus \Psi_\theta \neq \emptyset \\ \delta(\theta) & \text{otherwise.} \end{cases} \end{aligned}$$

Then  $\delta(\theta)$  and  $\Delta(\theta)$  each depend continuously on  $\theta$ . Moreover, it is straightforward to show that  $\Delta(\theta_i) > \delta(\theta_i) = 0$ , since we have assumed  $|\Theta(A)| \geq 2$  (this condition could only ever be violated if all elements of  $A$  were scalar multiples of one another). By continuity, it follows that for  $|\theta - \theta_i|$  sufficiently small we have  $\delta(\theta) < \Delta(\theta)$ , which means that  $\Psi_\theta \subseteq \Psi_{\theta_i}$ . But for  $|\theta - \theta_i| \leq \frac{\pi}{2}$  and for  $\mathbf{y} \in \Psi_{\theta_i}$ , we have the explicit formula

$$|\mathbf{g}_\theta^\perp \cdot \mathbf{y}| = \|\mathbf{y}\| \sin |\theta - \theta_i|.$$

This is obviously minimized by whichever  $\mathbf{y}^* \in \Psi_{\theta_i}$  is shortest - i.e.  $\|\mathbf{y}^*\| \leq \|\mathbf{y}\|$  for all  $\mathbf{y} \in \Psi_{\theta_i}$ . Moreover, since  $A$  is contained in the lower half plane we know this



**Fig. 28** Proving that the point casting the shallowest angle on  $L_g$  from above is also the point minimizing the orthogonal distance from above: Given  $y_i \in A \subseteq \mathbb{Z}^2 \cap \{y \leq -1\}$  and line  $L_g = \{\lambda g : \lambda \in \mathbb{R}\}$ ,  $g = (\cos \theta, \sin \theta)$  passing through the origin, we define the (open) triangles  $T_i, \tilde{T}_i$  to be the unique pair of open triangles with one side parallel to  $L_g$ , another side horizontal, and a third side equal to the ray from the origin to  $y_i$ . A symmetry-based argument in Proposition 5 shows that, under modest hypotheses on  $A$ , the triangle  $T_i$  contains a lattice point (element of  $\mathbb{Z}^2$ ) if and only if  $\tilde{T}_i$  does

minimizer is unique. Hence  $\Psi_\theta = \{y^*\}$ , which makes  $y^*$  a singleton minimizer. Since  $\theta_i = \theta(y^*)$ , this proves the desired inclusion (75), and we have already proven our secondary claim on the length of  $y^*$  being minimal.  $\square$

Our next claim was a formula for  $\Psi$  valid when  $\theta_i < \theta < \theta_{i+1}$  for two consecutive members  $\theta_i, \theta_{i+1} \in \Theta(A)$ , when  $0 := \theta_{0,1} < \theta < \theta_1$ , and when  $\theta_n < \theta < \theta_{n,n+1} := \pi$ . Proposition 5 derives this formula, under the assumption that  $A$  can be described a union of discrete rectangles on or below the line  $y = -1$  and straddling the line  $x = 0$ . This includes the case  $A = b_r^-$ , but also covers a number of other cases, as mentioned in Remark 19. Credit for this proposition goes to Dömötör Pálvölgyi, who had the critical idea of using a symmetry based argument [18].

**Proposition 5** *Let  $A \subseteq \mathbb{Z}^2 \cap \{y \leq -1\}$  be a finite union of discrete rectangles of the form*

$$A = \bigcup_{k=1}^K [a_k, b_k] \times [c_k, d_k] \cap \mathbb{Z}^2$$

where  $-\infty < a_k \leq 0 \leq b_k < \infty$ ,  $-\infty < c_k \leq d_k \leq -1$  for all  $k$ . Let

$$\Theta(A) := \{\theta_1, \theta_2, \dots, \theta_n\}$$

denote the angular spectrum of  $A$ , let  $g = (\cos \theta, \sin \theta)$ , and let  $Y = \{y_1, y_2, \dots, y_n\}$  be the set of singleton minimizers defined in Proposition 4 of  $|g^\perp \cdot y|$  over  $A$  as  $\theta$  varies from 0 to  $\pi$ . For each  $1 \leq i \leq n-1$ , define the transition angle  $\theta_{i,i+1} \in (\theta_i, \theta_{i+1})$  by

$$\theta_{i,i+1} = \theta(y_i + y_{i+1}).$$

Define also  $\theta_{0,1} := 0$  and  $\theta_{n,n+1} = \pi$  for convenience. Then

$$\Psi = \begin{cases} \{\mathbf{y}_i\} & \text{if } \theta_i < \theta < \theta_{i,i+1} \text{ for some } i = 1, \dots, n \\ \{\mathbf{y}_i, \mathbf{y}_{i+1}\} & \text{if } \theta = \theta_{i,i+1} \text{ for some } i = 1, \dots, n-1 \\ \{\mathbf{y}_{i+1}\} & \text{if } \theta_{i,i+1} < \theta < \theta_{i+1} \text{ for some } i = 0, \dots, n-1 \end{cases}$$

**Proof** The bulk of the work is to prove that if  $\theta_i < \theta < \theta_{i+1}$ , then

$$\Psi_\theta := \operatorname{argmin}_{\mathbf{y} \in A} |\mathbf{g}^\perp \cdot \mathbf{y}| \subseteq \{\mathbf{y}_i, \mathbf{y}_{i+1}\}.$$

Once this is established, since we evidently have  $\Psi_\theta = \{\mathbf{y}_i\}$ ,  $\Psi_\theta = \{\mathbf{y}_{i+1}\}$  for  $\theta$  sufficiently close to  $\theta_i$  and  $\theta_{i+1}$  respectively, it follows that that

$$\Psi_\theta = \begin{cases} \{\mathbf{y}_i\} & \text{if } \theta < \theta_c \\ \{\mathbf{y}_i, \mathbf{y}_{i+1}\} & \text{if } \theta = \theta_c \\ \{\mathbf{y}_{i+1}\} & \text{if } \theta > \theta_c. \end{cases}$$

where  $\theta_c$  is defined by  $|\mathbf{g}^\perp \cdot \mathbf{y}_i| = |\mathbf{g}^\perp \cdot \mathbf{y}_{i+1}|$ . One can readily show this is equivalent to  $\theta_c = \theta(\mathbf{y}_i + \mathbf{y}_{i+1})$ .

To prove that  $\Psi_\theta := \operatorname{argmin}_{\mathbf{y} \in A} |\mathbf{g}^\perp \cdot \mathbf{y}| \subseteq \{\mathbf{y}_i, \mathbf{y}_{i+1}\}$  as claimed, consider the open triangles  $T_i, \tilde{T}_i$  defined in terms of  $\mathbf{y}_i$  as shown in Fig. 28, as well as open triangles  $T_{i+1}, \tilde{T}_{i+1}$  defined in the same way in terms of  $\mathbf{y}_{i+1}$ . The triangles  $T_i$  and  $T_{i+1}$  each have empty intersection with  $A$ , as  $\mathbf{y}_i$  and  $\mathbf{y}_{i+1}$  are the elements of  $A$  that cast the shallowest angles on  $L_{\mathbf{g}}$  from above and below. To prove  $\Psi_\theta \subseteq \{\mathbf{y}_i, \mathbf{y}_{i+1}\}$ , we need to show that  $\mathbf{y}_i$  and  $\mathbf{y}_{i+1}$  are also the two *closest* points in  $A$  to  $L_{\mathbf{g}}$ . This amounts to proving that the triangles  $\tilde{T}_i$  and  $\tilde{T}_{i+1}$  have empty intersection with  $A$  as well.

To show this, first note that our assumptions on  $A$  imply that the intersection of each of our four triangles with  $A$  is equal to their intersection with  $\mathbb{Z}^2$  as a whole (because  $A$  has no “holes”). Second, note that the map

$$\mathbf{F}(\mathbf{x}) = \mathbf{y}_i - \mathbf{x}$$

is a bijection of the plane taking  $T_i$  to  $\tilde{T}_i$  such that  $\mathbf{F}(\mathbb{Z}^2) = \mathbb{Z}^2$ . Hence  $\tilde{T}_i$  contains a lattice point if and only if  $T_i$  does. But

$$T_i \cap \mathbb{Z}^2 = T_i \cap A = \emptyset$$

by assumption, and so

$$\tilde{T}_i \cap \mathbb{Z}^2 = \tilde{T}_i \cap A = \emptyset$$

as well. This proves the claim for  $\tilde{T}_i$  and the proof for  $\tilde{T}_{i+1}$  is analogous. The remaining cases  $0 := \theta_{0,1} < \theta < \theta_1$  and  $\theta_n < \theta < \theta_{n,n+1} := \pi$  are straightforward and left as an exercise.  $\square$

Proposition 5 has a couple of straightforward corollaries. The first is our claim from 7.1.2 that  $\Psi$  is a singleton set for all but finitely many  $\theta$ . This is obvious from the statement of the proposition (which gives an expression for  $\Psi$  for all but finitely many  $\theta$ ) and requires no proof. The second corollary generalizes our formula for  $\theta_r^*$  from Sect. 7.1.2, and uses the following observation, which we also used in Sect. 7.1.2 and owe a proof of.

**Observation 7** *Let  $\mathbf{v}, \mathbf{w}$  be unit vectors in  $S^1$ . Then*

$$\theta(\mathbf{v} + \mathbf{w}) = \frac{\theta(\mathbf{v}) + \theta(\mathbf{w})}{2}.$$

**Proof** This is simplest if we work in complex arithmetic, that is, we write  $\mathbf{v} = e^{i\psi}$  and  $\mathbf{w} = e^{i\phi}$  for some  $\psi, \phi \in [0, 2\pi)$ . However, by symmetry we may assume  $\mathbf{v} = 1$  (otherwise rotate the plane). Hence, it suffices to prove

$$\frac{1 + e^{i\phi}}{|1 + e^{i\phi}|} = e^{i\frac{\phi}{2}}.$$

But this follows from the following simple manipulation:

$$1 + e^{i\phi} = e^{i\frac{\phi}{2}}(e^{-i\frac{\phi}{2}} + e^{i\frac{\phi}{2}}) = 2 \cos\left(\frac{\phi}{2}\right) e^{i\frac{\phi}{2}}.$$

$\square$

The following corollary shows that coherence transport-like algorithms, which use the same weights but replace  $B_{\epsilon,h}(\mathbf{x})$  with a different set of pixels (a finite union of discrete rectangles) exhibit similar kinking behaviour in the limit  $\mu \rightarrow \infty$ .

**Corollary 2** *Suppose we inpaint  $D = (0, 1]^2$  using Algorithm 1 with boundary data  $u_0 : \mathcal{U} \rightarrow \mathbb{R}^d$  and suppose the symmetry assumptions of Sect. 6.1 hold as usual. Assume our stencil  $a_r^*$  is of the form*

$$a_r^* = \bigcup_{k=1}^K [a_k, b_k] \times [c_k, d_k] \cap \mathbb{Z}^2$$

where  $-\infty < a_k \leq 0 \leq b_k < \infty$ ,  $c_k \leq d_k \leq -1$  for all  $k$ . Let

$$\Theta(a_r^*) := \{\theta_1, \theta_2, \dots, \theta_n\}$$

denote the angular spectrum of  $a_r^*$ , let  $\mathbf{g} = (\cos \theta, \sin \theta)$ , assume we use as stencil weights the weights of März (6), and denote by  $\mathbf{g}_{\mu,r}^*$  the limiting transport direction from Theorem 6. Let  $\mathbf{g}_r^* = \lim_{\mu \rightarrow \infty} \mathbf{g}_{\mu,r}^*$  and define  $\theta_r^* := \theta(\mathbf{g}_r^*)$ . Then



$$\theta_r^* = \begin{cases} \frac{\pi}{2} & \text{if } \theta = 0 \\ \theta_i & \text{if } \theta_{i-1,i} < \theta < \theta_{i,i+1} \text{ for some } i = 1, \dots, n \\ \frac{\theta_i + \theta_{i+1}}{2} & \text{if } \theta = \theta_{i,i+1} \text{ for some } i = 1, \dots, n \end{cases} \quad (76)$$

**Proof** This follows from Proposition 5 and observation 7 in exactly the same way as when showed this for the special case  $a_r^* = b_r^-$  in Sect. 7.1.2.  $\square$

We conclude this appendix with a remark on a practical algorithm for computing the angular spectrum  $\Theta(A)$  given  $A \subseteq \mathbb{Z}^2$  satisfying the hypotheses of Proposition 4. This algorithm was used to generate the theoretical limiting curves for  $\theta_r^*$  for coherence transport in Sect. 7.1.2.

**Remark 20** Given  $A \subseteq \mathbb{Z}^2$  satisfying the hypotheses of Proposition 4, a simple algorithm for computing the angular spectrum  $\Theta(A)$  and singleton minimizers  $Y$  is as follows:

1. Starting with  $Y^* = \emptyset$ , go through each  $\mathbf{y} \in A$  and find the unique  $\mathbf{y}' \in A$  such that  $\theta(\mathbf{y}) = \theta(\mathbf{y}')$  and  $\mathbf{y}'$  is of minimal length. If  $\mathbf{y}'$  is not already in  $Y^*$ , add it.
2. For each  $\mathbf{y} \in Y^*$ , compute  $\theta(\mathbf{y})$ . Sort  $Y^*$  according to  $\theta(\mathbf{y})$ . The sorted list  $Y^*$  is now equal to  $Y$ , and the sorted list of angles is  $\Theta(A)$ .

## F Proof of Theorem 1 Statement II

First off, note that since  $V_i \leq 0$ ,  $m \leq N$ , and  $Y_\tau \geq -(r+2)$  the restriction of our domain in the  $y$ -direction from  $\mathbb{Z}$  to  $\{-(r+2), -(r+1), \dots, N\}$  makes no difference. Putting this together with our periodic boundary conditions, we find that our situation is equivalent to a random walk on  $\mathbb{Z}^2$  modulo the equivalence relation

$$(i, j) \sim (i + kN, j) \quad \text{for all } k \in \mathbb{Z}.$$

Denoting by  $\rho_{\mathbf{X}_\tau}^*$  the density of our stopped random walk on the periodic domain and  $\rho_{\mathbf{X}_\tau}$  the density of the original walk on  $\mathbb{Z}^2$  considered in [36], we clearly have

$$\rho_{\mathbf{X}_\tau}^*(i, j) = \sum_{k \in \mathbb{Z}} \rho_{\mathbf{X}_\tau}(i + kN, j). \quad (77)$$

Now, for fixed  $i$  let  $k^*$  denote the value of  $k$  minimizing  $|i + kN - \mu|$ , where

$$\mu = n - \frac{\mu_x}{\mu_y} m$$

denotes the mean (in the  $x$ -direction) of the asymptotic distribution of the original random walk on  $\mathbb{Z}^2$ . Let us define

$$\mu^* = \mu \mod N.$$

Then it is not hard to see that we have

$$|i + k^*N - \mu| = d_N^{\text{circ}}(i, \mu^*),$$

where  $d_N^{\text{circ}}$  is the circular distance (46).

With this in mind, let us rewrite (77) as

$$\rho_{\mathbf{X}_t}^*(i, j) = \sum_{\Delta k \in \mathbb{Z}} \rho_{\mathbf{X}_t}(i + (k^* + \Delta k)N, j).$$

By the first statement of Theorem 1 we have

$$\begin{aligned} \rho_{\mathbf{X}_t}^*(i, j) &= \sum_{\Delta k \in \mathbb{Z}} \Delta(j) \frac{e^{-\frac{(d_N^{\text{circ}}(i, \mu^*) + \Delta k N)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} + o\left(\frac{1}{\sqrt{m}}\right) \\ &= \Delta(j) \frac{e^{-\frac{d_N^{\text{circ}}(i, \mu^*)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} + R + o\left(\frac{1}{\sqrt{m}}\right) \end{aligned}$$

where

$$\begin{aligned} R &= \sum_{\Delta k \in \mathbb{Z} \setminus \{0\}} \Delta(j) \frac{e^{-\frac{(d_N^{\text{circ}}(i, \mu^*) + \Delta k N)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} \\ &\leq \sum_{\Delta k \in \mathbb{Z} \setminus \{0\}} \frac{e^{-\frac{(d_N^{\text{circ}}(i, \mu^*) + \Delta k N)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)}. \end{aligned}$$

One may easily show that  $d_N^{\text{circ}}(i, \mu^*) \leq \frac{N}{2}$ . It follows that for  $\Delta k \in \mathbb{Z} \setminus \{0\}$  we have

$$\begin{aligned} (d_N^{\text{circ}}(i, \mu^*) + \Delta k N)^2 &\geq \left(|\Delta k|N - \frac{N}{2}\right)^2 \\ &= \left(\Delta k^2 - |\Delta k| + \frac{1}{4}\right)N^2. \end{aligned}$$

We therefore have:

$$\begin{aligned} R &= \sum_{\Delta k \in \mathbb{Z} \setminus \{0\}} \frac{e^{-\frac{(d_N^{\text{circ}}(i, \mu^*) + \Delta k N)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} \\ &\leq \sum_{\Delta k \in \mathbb{Z} \setminus \{0\}} \frac{e^{-\frac{(\Delta k^2 - |\Delta k| + \frac{1}{4})N^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} \end{aligned}$$

However, the mapping  $\Delta k \in \mathbb{Z} \setminus \{0\} \rightarrow \Delta k^2 - |\Delta k|$  takes values in  $\mathbb{N} \cup \{0\}$  and is two to one. Therefore

$$\begin{aligned} R &\leq 2 \sum_{k=0}^{\infty} \frac{e^{-\frac{(k+\frac{1}{4})N^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} \leq \frac{2}{\sqrt{2\pi}\sigma(m)} e^{-\frac{N^2}{8\sigma(m)^2}} \sum_{k=0}^{\infty} \left( e^{-\frac{N^2}{2\sigma(m)^2}} \right)^k \\ &= \frac{2}{\sqrt{2\pi}\sigma(m)} \frac{e^{-\frac{N^2}{8\sigma(m)^2}}}{1 - e^{-\frac{N^2}{2\sigma(m)^2}}} \leq \frac{2}{\sqrt{2\pi}\sigma(m)} e^{-\frac{N^2}{8\sigma(m)^2}} \end{aligned}$$

Now,  $\sigma(m)^2 = \tilde{\sigma}^2 m$  where  $\tilde{\sigma}^2 = \frac{y^2}{|\mu_y|^3}$  is a constant. Hence, for  $m$  sufficiently large we have  $\sigma(m) \geq 1$ . Noting that  $m = yN$ , for  $m$  sufficiently large we have

$$R \leq e^{-\frac{N^2}{8\sigma(m)^2}} = e^{-\frac{y^2}{8\tilde{\sigma}^2} m}.$$

Hence

$$\begin{aligned} \rho_{\mathbf{x}_\tau}^*(i, j) &\leq \frac{e^{-\frac{d_N^{\text{circ}}(i, \mu^*)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} + e^{-\frac{y^2}{8\tilde{\sigma}^2} m} + o\left(\frac{1}{\sqrt{m}}\right) \\ &= \frac{e^{-\frac{d_N^{\text{circ}}(i, \mu^*)^2}{2\sigma(m)^2}}}{\sqrt{2\pi}\sigma(m)} + o\left(\frac{1}{\sqrt{m}}\right). \end{aligned}$$

## G Proof of Lemma 1 Statement II

The proof follows from the first statement of Lemma 1 together with with routine tail estimates for Gaussians and other simple manipulations. First, note that

$$\begin{aligned} &|G_{\sigma(m), \mu, N}^*(x) - g_{\sigma(m), \mu, N}^*(x)| \\ &= \left| e^{-\frac{d_N^{\text{circ}}(x, \mu)^2}{2\sigma(m)^2}} \left| \frac{1}{\sqrt{2\pi}\sigma(m)} - \frac{1}{\sum_{n=0}^{N-1} e^{-\frac{d_N^{\text{circ}}(n, \mu)^2}{2\sigma(m)^2}}} \right| \right| \\ &\leq \frac{1}{\sqrt{2\pi}\sigma(m) \sum_{n=0}^{N-1} e^{-\frac{d_N^{\text{circ}}(n, \mu)^2}{2\sigma(m)^2}}} \left| \sqrt{2\pi}\sigma(m) - \sum_{n=0}^{N-1} e^{-\frac{d_N^{\text{circ}}(n, \mu)^2}{2\sigma(m)^2}} \right| \end{aligned}$$

Let us define

$$\mathcal{E} = \{d_N^{\text{circ}}(n, \mu) : n = 0, \dots, N-1\}.$$

We do not need to characterize  $\mathcal{E}$  precisely. It is enough for us to note that for  $N$  sufficiently large we have

$$\mathcal{E} \supseteq \left\{ -\left(\hat{\mu} - \left\lfloor \frac{N}{4} \right\rfloor - 1\right), \dots, -(\hat{\mu} - 1), \hat{\mu}, \hat{\mu} + 1, \dots, \hat{\mu} + \left\lceil \frac{N}{4} \right\rceil + 1 \right\},$$

where  $\hat{\mu} := \mu - \lfloor \mu \rfloor$ . From this it follows that

$$\sum_{n=0}^{N-1} e^{-\frac{d_{N+1}^{\text{circ}}(n, \mu)^2}{2\sigma(m)^2}} \geq \sum_{n=-\lfloor \frac{N}{4} \rfloor - 1}^{\lceil \frac{N}{4} \rceil + 1} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} \quad (78)$$

First, by (78) together with the assumption  $\sigma(m)^2 = \tilde{\sigma}^2 m \rightarrow \infty$  as  $m \rightarrow \infty$  we have

$$\begin{aligned} \sqrt{2\pi}\sigma(m) \sum_{n=0}^{N-1} e^{-\frac{d_{N+1}^{\text{circ}}(n, \mu)^2}{2\sigma(m)^2}} &\geq \sqrt{2\pi}\sigma(m) \sum_{n=-\lfloor \frac{N}{4} \rfloor - 1}^{\lceil \frac{N}{4} \rceil + 1} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} \\ &\geq \sqrt{2\pi}\sigma(m) e^{-\frac{\hat{\mu}^2}{2\sigma(m)^2}} \\ &\geq 1 \quad \text{for } m \text{ sufficiently large.} \end{aligned}$$

Hence, for  $m$  sufficiently large we have:

$$\begin{aligned} |G_{\sigma(m), \mu, N}^*(x) - g_{\sigma(m), \mu, N}^*(x)| &\leq \left| \sqrt{2\pi}\sigma(m) - \sum_{n=0}^{N-1} e^{-\frac{d_{N+1}^{\text{circ}}(n, \mu)^2}{2\sigma(m)^2}} \right| \\ &\leq \left| \sqrt{2\pi}\sigma(m) - \sum_{n \in \mathbb{Z}} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} \right| \\ &\quad + \left| \sum_{n \in \mathbb{Z}} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} - \sum_{n=0}^{N-1} e^{-\frac{d_{N+1}^{\text{circ}}(n, \mu)^2}{2\sigma(m)^2}} \right| \\ &\leq C\tilde{\sigma}\sqrt{m}e^{-2\pi^2\tilde{\sigma}^2 m} \\ &\quad + \left| \sum_{n \in \mathbb{Z}} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} - \sum_{n=-\lceil \frac{N}{4} \rceil + 1}^{\lfloor \frac{N}{4} \rfloor - 1} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} \right| \end{aligned}$$

where we have used Lemma 1 part one and (78) for the third inequality. Next,

$$\begin{aligned}
 & \left| \sum_{n \in \mathbb{Z}} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} - \sum_{n=-\lceil \frac{N}{4} \rceil-1}^{\lfloor \frac{N}{4} \rfloor+1} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} \right| \\
 &= \sum_{n=-\infty}^{-\lfloor \frac{N}{4} \rfloor-2} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} + \sum_{n=\lceil \frac{N}{4} \rceil+2}^{\infty} e^{-\frac{(n-\hat{\mu})^2}{2\sigma(m)^2}} \\
 &\leq \int_{-\infty}^{-\frac{N}{4}} e^{-\frac{x^2}{2\sigma(m)^2}} dx + \int_{\frac{N}{4}}^{\infty} e^{-\frac{x^2}{2\sigma(m)^2}} dx \\
 &= 2 \int_{\frac{N}{4}}^{\infty} e^{-\frac{x^2}{2\sigma(m)^2}} dx \leq \int_{\frac{N}{4}}^{\infty} \frac{x}{N/4} e^{-\frac{x^2}{2\sigma(m)^2}} dx \\
 &= \frac{\sigma(m)^2}{N/4} e^{-\frac{N^2}{32\sigma(m)^2}} \\
 &= 4\tilde{\sigma}^2 y e^{-\frac{y^2 m}{32\tilde{\sigma}^2}}
 \end{aligned}$$

where we have use  $m = yN$  in the last equality. This gives:

$$\begin{aligned}
 |G_{\sigma(m), \mu, N}^*(x) - g_{\sigma(m), \mu, N}^*(x)| &\leq C\tilde{\sigma}\sqrt{m}e^{-2\pi^2\tilde{\sigma}^2 m} + 4\tilde{\sigma}^2 y e^{-\frac{y^2 m}{32\tilde{\sigma}^2}} \\
 &\leq (C\tilde{\sigma} + 4\tilde{\sigma}^2 y)\sqrt{m}e^{-\min(2\pi^2\tilde{\sigma}^2, \frac{y^2}{32\tilde{\sigma}^2})m}
 \end{aligned}$$

From which the claim follows with  $A = C\tilde{\sigma} + 4\tilde{\sigma}^2 y$  and  $b = \min(2\pi^2\tilde{\sigma}^2, \frac{y^2}{32\tilde{\sigma}^2})$ .

## References

1. Ambrosio, L., Fusco, N., Pallara, D.: Functions of bounded variation and free discontinuity problems. Oxford Mathematical Monographs (2000)
2. Apostol, T.M.: Mathematical Analysis. Pearson (1974)
3. Arias, P., Facciolo, G., Caselles, V., Sapiro, G.: A variational framework for exemplar-based image inpainting. *Int. J. Comput. Vision* **93**(3), 319–347 (2011). <https://doi.org/10.1007/s11263-010-0418-7>.
4. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., Verdera, J.: Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing* **10**(8), 1200–1211 (2001). <https://doi.org/10.1109/83.935036>
5. Bellman, R.: A Brief Introduction to Theta Functions. Holt (1961)
6. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: Proceedings of the 27th annual conference on Computer graphics and interactive techniques, pp. 417–424. ACM Press/Addison-Wesley Publishing Co. (2000)
7. Bornemann, F., März, T.: Fast image inpainting based on coherence transport. *Journal of Mathematical Imaging and Vision* **28**(3), 259–278 (2007). <https://doi.org/10.1007/s10851-007-0017-6>.
8. Bracewell, R.: The Fourier Transform and its Applications, second edn. McGraw-Hill Kogakusha, Ltd., Tokyo (1978)

9. Briggs, W.L., Henson, V.E., McCormick, S.F.: A Multigrid Tutorial: Second Edition. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2000)
10. Brown, R.A.: Barycentric coordinates as interpolants. CoRR (2013). [arXiv:1308.1279](https://arxiv.org/abs/1308.1279)
11. Burger, M., He, L., Schönlieb, C.: Cahn-hilliard inpainting and a generalization for grayvalue images. *SIAM J. Imaging Sci.* **2**(4), 1129–1167 (2009)
12. Buysens, P., Meur, O., Daisy, M., Tschumperlé, D., Lézoray, O.: Depth-guided disocclusion inpainting of synthesized rgb-d images. *IEEE Transactions on Image Processing* **26**(2), 525–538 (2017). <https://doi.org/10.1109/TIP.2016.2619263>
13. Chan, T., Kang, S., Shen, J.: Euler's elastica and curvature-based inpainting. *SIAM Journal on Applied Mathematics* pp. 564–592 (2002)
14. Chan, T., Shen, J.: Variational image inpainting. *Communications on pure and applied mathematics* **58**(5), 579–619 (2005)
15. Cilleruelo, J.: The distribution of the lattice points on circles. *Journal of Number Theory* **43**(2), 198 – 202 (1993). <https://doi.org/10.1006/jnth.1993.1017>
16. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing* **13**, 1200–1212 (2004)
17. Daribo, I., Pesquet-Popescu, B.: Depth-aided image inpainting for novel view synthesis. In: 2010 IEEE International Workshop on Multimedia Signal Processing, pp. 167–170 (2010). <https://doi.org/10.1109/MMSP.2010.5662013>
18. domotorp (<http://mathoverflow.net/users/955/domotorp>): Conjecture regarding closest point inside a discrete ball to a line. MathOverflow. <http://mathoverflow.net/q/187830> (version: 2014-11-22)
19. Erdős, P., Hall, R.: On the angular distribution of gaussian integers with fixed norm. *Discrete Mathematics* **200**(1–3), 87 – 94 (1999). [https://doi.org/10.1016/S0012-365X\(98\)00329-X](https://doi.org/10.1016/S0012-365X(98)00329-X)
20. Getreuer, P.: A Survey of Gaussian Convolution Algorithms. *Image Processing On Line* **3**, 286–310 (2013). <https://doi.org/10.5201/ipol.2013.87>
21. Guillemot, C., Meur, O.: Image inpainting : Overview and recent advances. *IEEE Signal Processing Magazine* **31**(1), 127–144 (2014). <https://doi.org/10.1109/MSP.2013.2273004>
22. Gut, A.: On the moments and limit distributions of some first passage times. *Ann. Probab.* **2**(2), 277–308 (1974). <https://doi.org/10.1214/aop/1176996709>
23. Gut, A.: On the moments of some first passage times for sums of dependent random variables. *Stochastic Processes and their Applications* **2**(1), 115 – 126 (1974). [https://doi.org/10.1016/0304-4149\(74\)90015-5](https://doi.org/10.1016/0304-4149(74)90015-5). <http://www.sciencedirect.com/science/article/pii/0304414974900155>
24. Gut, A.: Stopped Random Walks: Limit Theorems and Applications. Springer Series in Operations Research and Financial Engineering. Springer New York (2009). <https://books.google.co.uk/books?id=WfBs5G7jHIC>
25. Gut, A., Janson, S.: The limiting behaviour of certain stopped sums and some applications. *Scandinavian Journal of Statistics* **10**(4), 281–292 (1983). <http://www.jstor.org/stable/4615930>
26. Hocking, L., Holding, T., Schönlieb, C.: Numerical analysis of shell-based geometric image inpainting algorithms and their semi-implicit extension. [arXiv:1707.09713](https://arxiv.org/abs/1707.09713)
27. Hocking, L., MacKenzie, R., Schönlieb, C.: Guidefill: Gpu accelerated, artist guided geometric inpainting for 3d conversion of film. [arXiv:1611.05319](https://arxiv.org/abs/1611.05319)
28. LeVeque, R.J.: Numerical methods for conservation laws (2. ed.). Lectures in mathematics. Birkhäuser (1992)
29. Ma, L., Do, L., de With, P.: Depth-guided inpainting algorithm for free-viewpoint video. In: 2012 19th IEEE International Conference on Image Processing, pp. 1721–1724 (2012). <https://doi.org/10.1109/ICIP.2012.6467211>
30. Malinovskii, V.K.: Limit theorems for stopped random sequences. i: Rates of convergence and asymptotic expansions. *Theory of Probability & Its Applications* **38**(4), 673–693 (1994). <https://doi.org/10.1137/1138067>
31. März, T.: Image inpainting based on coherence transport with adapted distance functions. *SIAM J. Img. Sci.* **4**(4), 981–1000 (2011). <https://doi.org/10.1137/100807296>
32. März, T.: A well-posedness framework for inpainting based on coherence transport. *Foundations of Computational Mathematics* **15**(4), 973–1033 (2015). <https://doi.org/10.1007/s10208-014-9199-7>
33. Masnou, S., Morel, J.: Level lines based disocclusion. In: Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on, pp. 259–263. IEEE (1998)
34. Schönlieb, C.: Partial Differential Equation Methods for Image Inpainting. Cambridge University Press (2015)

35. Smith, S.W.: The Scientist and Engineer's Guide to Digital Signal Processing. California Technical Publishing, San Diego, CA, USA (1997)
36. Stam, A.J.: Local central limit theorem for first entrance of a random walk into a half space. *Compositio Mathematica* **23**(1), 15–23 (1971). <http://eudml.org/doc/89073>
37. Tao, T.: Variants of the central limit theorem. <https://terrytao.wordpress.com/2015/11/19/275a-notes-5-variants-of-the-central-limit-theorem/#more-8566>
38. Telea, A.: An image inpainting technique based on the fast marching method. *Journal of Graphics Tools* **9**(1), 23–34 (2004)
39. Trottenberg, U., Oosterlee, C., Schuller, A.: Multigrid. Elsevier Science (2000). <https://books.google.ca/books?id=9ysyNPZoR24C>
40. TschumperlÉ, D.: Fast anisotropic smoothing of multi-valued images using curvature-preserving pde's. *International Journal of Computer Vision* **68**(1), 65–82 (2006). <https://doi.org/10.1007/s11263-006-5631-z>.
41. Varah, J.: A lower bound for the smallest singular value of a matrix. *Linear Algebra and its Applications* **11**(1), 3 – 5 (1975). [https://doi.org/10.1016/0024-3795\(75\)90112-3](https://doi.org/10.1016/0024-3795(75)90112-3). <http://www.sciencedirect.com/science/article/pii/0024379575901123>
42. Varga, R.: Matrix iterative analysis. Prentice-Hall series in automatic computation. Prentice-Hall (1962). <https://books.google.co.uk/books?id=PFYWAAAIAAJ>
43. Wexler, Y., Shechtman, E., Irani, M.: Space-time completion of video. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(3), 463–476 (2007). <https://doi.org/10.1109/TPAMI.2007.60>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## Affiliations

L. Robert Hocking<sup>1</sup> · Thomas Holding<sup>2</sup> · Carola-Bibiane Schönlieb<sup>1</sup>

Thomas Holding  
t.holding@warwick.ac.uk

Carola-Bibiane Schönlieb  
cbs31@cam.ac.uk

<sup>1</sup> Department for Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge CB3 0WA, UK

<sup>2</sup> Mathematics Institute, Zeeman Building, University of Warwick, Coventry CV4 7AL, UK