SIAM J. SCI. COMPUT.
Vol. 41, No. 5, pp. A3097–A3117

# IMPROVED ACCURACY OF MONOTONE FINITE DIFFERENCE SCHEMES ON POINT CLOUDS AND REGULAR GRIDS[*]

CHRIS FINLAY[†] AND ADAM OBERMAN[†]

**Abstract.** Finite difference schemes are the method of choice for solving nonlinear, degenerate elliptic PDEs, because the Barles–Souganidis convergence framework [G. Barles and P. E. Souganidis, *Asymptotic Anal.*, 4 (1991), pp. 271–283] provides sufficient conditions for convergence to the unique viscosity solution [M. G. Crandall, H. Ishii, and P.-L. Lions, *Bull. Amer. Math. Soc. (N.S.)*, 27 (1992), pp. 1–67]. For anisotropic operators, such as the Monge–Ampère equation, wide stencil schemes are needed [A. M. Oberman, *SIAM J. Numer. Anal.*, 44 (2006), pp. 879–895]. The accuracy of these schemes depends on both the distances to neighbors, $R$, and the angular resolution, $d\theta$. On regular grids, the accuracy is $\mathcal{O}(R^2 + d\theta)$. On point clouds, the most accurate schemes are of $\mathcal{O}(R + d\theta)$, by Froese [*Numer. Math.*, 138 (2018), pp. 75–99]. In this work, we construct geometrically motivated schemes of higher accuracy in both cases: order $\mathcal{O}(R + d\theta^2)$ on point clouds, and $\mathcal{O}(R^2 + d\theta^2)$ on regular grids.

**Key words.** finite differences, nonlinear degenerate elliptic partial differential equations, stability and convergence

**AMS subject classifications.** 65N06, 65N12, 35J15, 35J60, 35J70

**DOI.** 10.1137/18M1200269

## 1. Introduction.

The goal of this paper is to build more accurate convergent discretizations for the class of nonlinear elliptic PDEs [CIL92]. Our schemes are applicable in both two and three dimensions for a class of PDEs which includes the convex envelope operator and the Pucci operator, as well as the Monge–Ampère operator. Convergent discretizations for these second order operators are available on regular grids [Obe08b], but the accuracy of these schemes depends on both the distances to neighbors, $R$, and the angular resolution, $d\theta$. On regular grids, the accuracy (the discretization error of the operator) is $\mathcal{O}(R^2 + d\theta)$. More recently, Froese [Fro18] developed methods on point clouds of accuracy $\mathcal{O}(R + d\theta)$. These schemes were used for freeform optical design to shape laser beams [FFL$^+$17], an application which required nonregular grids. In this work, we construct geometrically motivated schemes of higher accuracy in both cases: order $\mathcal{O}(R + d\theta^2)$ on point clouds, and $\mathcal{O}(R^2 + d\theta^2)$ on regular grids.

Even higher accuracy is possible when the operator is uniformly elliptic. For example, in the set of papers [BCM16, FM14, Mir14a, Mir14b], Mirebeau and coauthors developed a framework for constructing $\mathcal{O}(h^2)$ monotone and stable schemes for several functions of the eigenvalues of the Hessian on regular grids in two dimensions. Related work for discretization of convex functions is studied in [Mir16]. Mirebeau studied monotone discretization of first order (Eikonal-type) equations on triangulated grids [Mir14a] as well as second order Monge–Ampère-type operators [Mir14b]. In the latter case, he obtains nearly optimal accuracy, but his construction is most effective when the operator is uniformly elliptic: as the operator degenerates, the width of the stencil increases. Moreover, the elegant construction based on the Stern–Brocot tree is particular to two dimensions.

[*] Submitted to the journal's Methods and Algorithms for Scientific Computing section July 13, 2018; accepted for publication (in revised form) July 24, 2019; published electronically October 11, 2019.
https://doi.org/10.1137/18M1200269

[†] Department of Mathematics and Statistics, McGill University, Montreal, QC, H3A 0B9 Canada (christopher.finlay@mail.mcgill.ca, adam.oberman@mcgill.ca).

Higher accuracy is also possible using filtered schemes [FO13, OS15, BPR16]. Filtered schemes combine a base monotone scheme with a higher accuracy scheme: however, increased accuracy of the base scheme is beneficial to the filtered scheme, since it allows for a smaller filter parameter.

The challenge of building monotone convergent finite difference schemes is illustrated in [CWL16, CW17], where the Monge–Ampère equation is discretized in two dimensions. In [CWL16], a mixture of a 7-point stencil for the cross derivative and a semi-Lagrangian wide stencil was used. The 7-point stencil was used for the cross derivative when it is monotone; otherwise the wide stencil was employed. This approach was later employed in a coarsening strategy for multigrid in [CW17] but does not fully solve the problem of building narrow monotone stencils and has not been generalized to higher dimensions.

Another approach lies between the wide stencil finite difference approach and the finite element approach. In [NNZ19], a convergent method on an unstructured mesh is constructed on two separate scales, specifically for the Monge–Ampère equation. In this work, the authors of [NNZ19] prove the convergence of their method while disentangling dependence between the spatial and directional resolution parameters. The focus of [NNZ19] was on the asymptotic convergence of their method. In contrast, our method aims for high accuracy *given a fixed grid size*, which is more consistent with computational restrictions. Our method is similar to [NNZ19] in that we also use linear interpolation to construct our finite difference scheme. The two-scale method of [NNZ19] uses interpolation of an $n$-dimensional simplex and is consistent on interior grid points away from a strip of the boundary. Our method uses interpolation of $(n-1)$-dimensional simplices and is consistent on all interior points. For a recent review of current methods, see [NSZ17].

The need for wide stencils arises from the anisotropy of the operators. For isotropic operators, such as the Laplacian, or for operators whose second order anisotropy happens to align with the grid (essentially combinations of $u_{xx}$ and $u_{yy}$ terms), an adaptive quadtree grid discretization was developed in [OZ16]. An adaptive quadtree grid was combined with the $\mathcal{O}(R+d\theta)$ mesh-free method of Froese [Fro18] and filtered schemes [OS15, FO13] in [FS17].

The main idea of this work is based on locating the reference point within two triangles (in two dimensions) or simplices (in three or higher dimensions) and using barycentric coordinates [DB08, sect. 5.4, p. 595] to write down the discretization. For first order derivatives, only one simplex is needed. It is standard to write a gradient of a function based on linear interpolation; extending this to a directional derivative amounts to computing a dot product. However, for second directional derivatives, it is possible to use two simplices to compute a monotone discretization of the second directional derivative, with accuracy which depends on the relative sizes of the simplices.

### 1.1. Off-directional discretizations.

When the direction $w$ does not align with the grid, the $d\theta$ term appears in the expression for the finite difference accuracy. If $u$ is discretized on a regular grid, then one common approach is to choose the nearest grid direction $v_h$ to $w$, and take the finite difference along this approximate direction, as in [Obe08b]. In the symmetric case for the second derivative, the finite difference remains $\mathcal{O}(h^2)$ but picks up a directional resolution error $d\theta$. This directional resolution error is first order and is given as $d\theta = \arccos\langle w, v_h/\left\|v_h\right\|\rangle$. Overall, this approach is $\mathcal{O}(d\theta+R^2)$ accurate, where $R$ is the stencil radius. On a grid with spatial resolution $h$, one can show that for a desired angular resolution $d\theta$, the stencil radius

Table 1
*Comparison of the discretizations.*

| Scheme | Order | Optimal $d\theta$ | Formal accuracy | Comments |
|---|---|---|---|---|
| Nearest grid direction [Obe08b] | $\mathcal{O}(R^2 + d\theta)$ | $\mathcal{O}(h^{\frac{2}{3}})$ | $\mathcal{O}(h^{\frac{2}{3}})$ | Regular grids. Difficult implementation near boundaries. |
| Two-scale convergence [NNZ19] | $\mathcal{O}(R^2 + d\theta^2)$ | $\mathcal{O}(h^{\frac{1}{2}})$ | $\mathcal{O}(h)$ | $n$-d, for triangulations. Consistent away from boundary. |
| Froese [Fro18] | $\mathcal{O}(R + d\theta)$ | $\mathcal{O}(h^{\frac{1}{2}})$ | $\mathcal{O}(h^{\frac{1}{2}})$ | 2d, mesh-free. No difficulty at boundary. |
| Linear interpolant, symmetric | $\mathcal{O}(R^2 + d\theta^2)$ | $\mathcal{O}(h^{\frac{1}{2}})$ | $\mathcal{O}(h)$ | $n$-d, regular grids. No difficulty at boundary. |
| Linear interpolant, nonsymmetric | $\mathcal{O}(R + d\theta^2)$ | $\mathcal{O}(h^{\frac{1}{3}})$ | $\mathcal{O}(h^{\frac{2}{3}})$ | $n$-d, mesh-free. No difficulty at boundary. |

$R$ is $\mathcal{O}(\frac{h}{d\theta})$ (see, for example, [Fro18] for details). With optimal choice $d\theta = \left(2h^2\right)^{\frac{1}{3}}$, this scheme is therefore formally $\mathcal{O}(h^{\frac{2}{3}})$. Although appealing due to its simplicity, this scheme suffers some drawbacks. It is only appropriate on regular finite difference grids and encounters difficulties discretizing $u$ near the boundary of the domain.

Recent work by Froese [Fro18] treats the more general case where $u$ is discretized on a cloud of point $\mathcal{G}$. Froese presents a monotone finite difference scheme for the second derivative which is $\mathcal{O}(R + d\theta)$. The parameter $R$ is a search radius, which will be defined more precisely later. Set $h = \sup_{x\in\Omega} \min_{x_j\in\mathcal{G}} \|x - x_j\|$. Then (as in the previous method) for a desired angular resolution, $R$ is $\mathcal{O}\left(\frac{h}{d\theta}\right)$, and so with the optimal choice of $d\theta = \sqrt{h}$, the method is formally $\mathcal{O}\big(\sqrt{h}\big)$. Unfortunately, this scheme does not generalize easily to higher dimensions.

In what follows, we present a monotone and consistent finite difference scheme for the first and second derivatives which overcomes the deficiencies of the preceding two methods. For the second derivative, if the grid is not regular, our scheme has accuracy $\mathcal{O}(R + d\theta^2)$, or formally $\mathcal{O}(h^{\frac{2}{3}})$. Further in the regular case, the scheme is $\mathcal{O}(R^2 + d\theta^2)$ and is formally $\mathcal{O}(h)$. The method works in dimension two and higher and can be used on any set of discretization points, regular or otherwise. Under mild requirements on the spatial resolution of the domain boundary (see Lemma 3), the scheme can easily be implemented near the boundary of a domain as well, with an appropriate choice of boundary resolution (see Remark 1). In particular, the scheme easily handles Neumann boundary conditions on nonrectangular domains.

Using these schemes as building blocks, we build monotone, stable, and consistent schemes for nonlinear degenerate elliptic equations on arbitrary meshes.

Table 1 presents a summary of the second derivative schemes discussed in this paper.

**1.2. Directional discretizations.** The basic building blocks of our discretization are first and second order directional derivatives. This is in contrast to the work of Mirebeau, where two-dimensional shapes built up of triangles are chosen to match the ellipticity of the operator.

Write the first and second directional derivatives of a function $u$ in the direction $w$ (with $||w|| = 1$) as

$$u_w = \langle w, Du\rangle, \qquad u_{ww} = w^{\mathsf{T}} D^2 u w,$$

where $Du$ and $D^2u$ are the gradient and Hessian of $u$, respectively.

Define the forward difference in the direction $v$ by

$$\mathcal{D}_v u(x) = \frac{u(x+v) - u(x)}{|v|}.$$

The first order monotone finite difference schemes for $u_w$ in the directions $tw$ and $-tw$ are given by

$$\begin{aligned} \mathcal{D}_{tw} u(x) &= u_w(x) + \mathcal{O}(t), \\ \mathcal{D}_{-tw} u(x) &= u_w(x) + \mathcal{O}(t). \end{aligned}$$

The simplest finite difference scheme for $u_{ww}$ is the centered finite difference scheme

$$\begin{aligned} \frac{u(x+tw) - 2u(x) + u(x-tw)}{t^2} &= \frac{1}{t}\left[\mathcal{D}_{tw} u(x) + \mathcal{D}_{-tw} u(x)\right] \\ &= u_{ww}(x) + \mathcal{O}(t^2). \end{aligned} \tag{1}$$

The generalization to unequally spaced points is clear from (1):

$$\frac{2}{t_p + t_m}\left[\mathcal{D}_{t_p w} u(x) + \mathcal{D}_{-t_m w} u(x)\right] = u_{ww}(x) + \mathcal{O}(t_+),$$

where $t_+ = \max\{t_p, t_m\}$ (in general, the scheme is first order accurate, unless $t_p = t_m$).

**1.3. Directional finite differences using barycentric coordinates.** Suppose we want to compute $u_w(x_0)$ using values $u(x_i)$ which determine a simplex. Using linear interpolation, we can approximate the value of $u(x + t_p w)$ on the boundary of the simplex. A convenient expression for this value is given by using barycentric coordinates (see, for example, [DB08, sect. 5.4, p. 595]), which allows us to generalize (1).

Suppose $\mathcal{S}_m$ and $\mathcal{S}_p$ are the vertices of an $(n-1)$-dimensional simplex. Suppose further that

$$r \le \|x_0 - x_i\| \le R \qquad \text{for all } x_i \in \{\mathcal{S}_m, \mathcal{S}_p\}.$$

Suppose further that

$$\begin{aligned} x_p &= x_0 + t_p w \text{ is in the simplex determined by } \mathcal{S}_p, \\ x_m &= x_0 - t_m w \text{ is in the simplex determined by } \mathcal{S}_n \end{aligned}$$

for $t_m, t_p \in [r, R]$. Construct the corresponding linear interpolants $L_m$ and $L_p$:

$$\begin{aligned} L_p(x) &= \sum_{i \in \mathcal{S}_p} \lambda_p^i(x) u(x_i), \\ L_m(x) &= \sum_{i \in \mathcal{S}_m} \lambda_m^i(x) u(x_i). \end{aligned}$$

Here $\lambda_p(x)$ and $\lambda_m(x)$ are the barycentric coordinates in $\mathcal{S}_p$ and $\mathcal{S}_m$ respectively. The barycentric coordinates are easily constructed. Let $v_i^p = x_i - x_0$, $i \in \mathcal{S}_p$, and similarly define $v_i^m$. By assumption, all $v_i$'s satisfy $r \le ||v_i|| \le R$. Let $V_p$ be the matrix

$$V_p = \begin{bmatrix} v_1^p & v_2^p & \dots & v_n^p \end{bmatrix}.$$

Then $\lambda_p$ is given by solving

$$V_p \lambda_p = x.$$

The barycentric coordinates $\lambda_m$ for $\mathcal{S}_m$ are defined analogously. By virtue of convexity, if $x$ lies in the (relative) interior of a simplex, its barycentric coordinates are positive and sum to one.

Barycentric coordinates allow us to define the finite difference schemes for the first and second directional derivatives as follows.

Definition 1 (first derivative scheme). *The first derivative scheme takes two forms, upwind and downwind, respectively:*

$$\begin{aligned}
\mathcal{D}_w u(x_0) &:= \frac{1}{t_p}\left(L_p(x_0 + t_p w) - u(x_0)\right), \qquad t_p = \frac{1}{1^\intercal V_p^{-1} w}, \\
\mathcal{D}_{-w} u(x_0) &:= \frac{1}{t_m}\left(L_p(x_0 - t_m w) - u(x_0)\right), \qquad t_m = \frac{-1}{1^\intercal V_m^{-1} w}.
\end{aligned} \tag{2}$$

Definition 2 (second derivative scheme). *The second derivative scheme is defined as*

$$\mathcal{D}_{ww} u(x_0) = \frac{2\left(\mathcal{D}_w u(x_0) + \mathcal{D}_{-w} u(x_0)\right)}{t_p + t_m},$$

*with $t_p$ and $t_m$ as given above.*

Lemma 1 (monotone and stable). *The finite difference schemes of Definitions 1 and 2 are monotone and stable.*

*Proof.* By convexity, we are guaranteed that $0 \leq \lambda^i_{p,m} \leq 1$. Further, we have that both $\sum \lambda^i_p = \sum \lambda^i_m = 1$. This corresponds to a monotone discretization of the operator [Obe06]. □

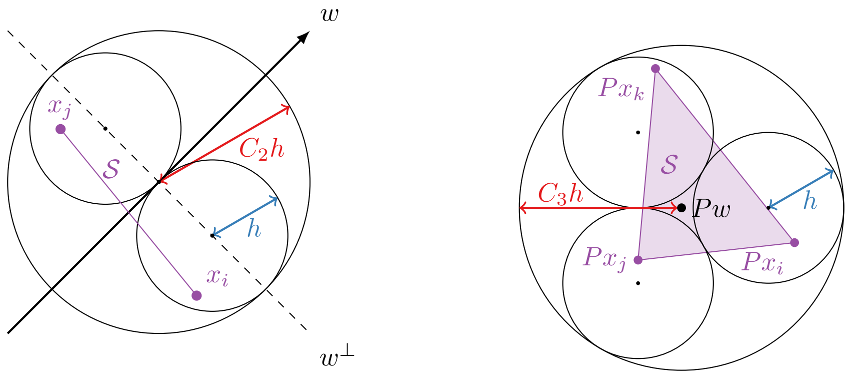
In the application below, we will use long, slender simplices, which are oriented near the directions $\pm w$, and control the interior and exterior radii, in order to establish the accuracy of the schemes.

## 2. The framework.

In this section, we introduce a framework for constructing monotone finite difference operators on a point cloud in two or three dimensions. To implement the method, we require finding triangles (in two dimensions) or tetrahedra (in three dimensions) which contain the reference point. The neighbors of the reference point form $(n-1)$-dimensional simplices. In practice, these simplices are found using an underlying triangulation, which could be provided by a mesh generator. (See Algorithm 1 in section 2.3 for details.) The configuration of these simplices determines the accuracy of the scheme.

### 2.1. Notation.

- $\Omega \subset \mathbb{R}^n$ is an open convex bounded domain with Lipshitz boundary $\partial\Omega$. We focus on the cases $n = 2$ and $n = 3$.
- $\mathcal{G} \subset \bar{\Omega}$ is a point cloud with points $x_i$, $i = 1 \dots N$.
- If $\mathcal{G}$ is given as the undirected graph of a triangulation, then $A$ is the corresponding adjacency matrix of the graph.
- $h = \sup_{x \in \Omega} \min_{y \in \mathcal{G}} \|x - y\|$ is the spatial resolution of the graph. Every ball of radius $h$ in $\bar{\Omega}$ contains at least one grid point.
- $h_B = \sup_{x \in \partial\Omega} \min_{y \in \mathcal{G} \cap \partial\Omega} \|x - y\|$ is the spatial resolution of the graph on the boundary.
- $\delta = \min_{x \in \mathcal{G} \cap \Omega} \min_{y \in \mathcal{G} \cap \partial\Omega} \|x - y\|$ is the minimum distance between an interior point and a boundary point.
- $\ell$ is the minimum length of all edges in the graph $\mathcal{G}$.



(a) $C_2 = 2$

(b) $C_3 = 1 + \frac{2}{\sqrt{3}}$

FIG. 1. *There exists an $n-1$ simplex $\mathcal{S}$ enclosing $w$, contained within ball of radius $C_n h$. In* 1b*, projections onto a plane perpendicular to $w$ are shown.*

- $d\theta$ is the desired angular resolution. We shall require at least $d\theta < \pi$.
- $R = C_n h(1 + \operatorname{cosec}(\frac{d\theta}{2}))$ is the maximal search radius and depends only on the angular resolution, the spatial resolution, and a constant $C_n$ determined by the dimension.
- $r = C_n h(-1 + \operatorname{cosec}(\frac{d\theta}{2}))$ is the minimal search radius. We will see that the minimal search radius is necessary to guarantee convergence of the schemes. Further, to guarantee the convergence of schemes near the boundary, it will be necessary to require $\delta \geq r$.
- $C_n$ is a constant determined by the dimension. In $\mathbb{R}^2$, $C_2 = 2$; in $\mathbb{R}^3$, $C_3 = 1 + \frac{2}{\sqrt{3}}$.

The construction of the schemes above requires the existence of simplices which intersect the vector $w$. For accuracy, we further require that the angular resolution of the simplices' diameter relative to the point $x_0$ is less than $d\theta$. The following three lemmas show that for given angular and spatial resolutions, such schemes exist. See Figure 1.

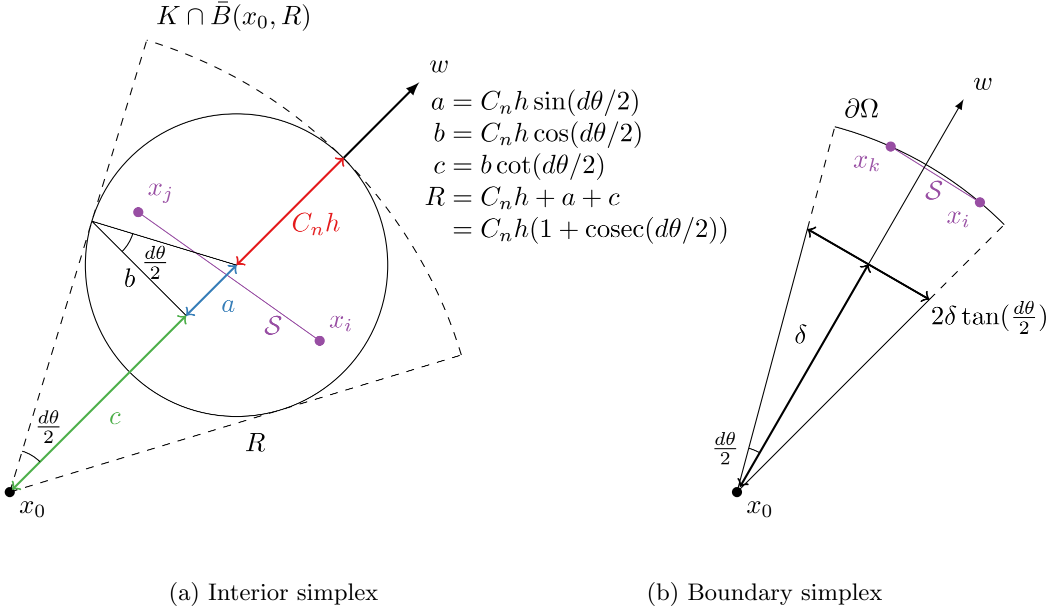
LEMMA 2 (existence of scheme away from boundary). *Take $x_0 \in \mathcal{G}$ having* $\operatorname{dist}(x_0, \partial\Omega) \geq R$. *Then it is possible to construct the simplices used in Definition* 1.

*Proof.* We must show that $\mathcal{S}_p$ and $\mathcal{S}_m$ exist. We first show the existence of the simplex $\mathcal{S}_p$; $\mathcal{S}_m$ follows similarly. Define the cone

$$K := \left\{ x \mid \frac{\langle v, w\rangle}{\|v\|} \geq 1 - \cos\left(\frac{d\theta}{2}\right), v = x - x_0 \right\}.$$

Any two points in $K$ have an angular resolution (relative to $x_0$) less than $d\theta$. Therefore, choosing points in this cone ensures the angular resolution is satisfied.

We must now show that the set $\mathcal{G} \cap K$ contains points defining $\mathcal{S}_p$. By construction, any ball in the interior of $\Omega$ with radius $h$ contains at least one interior point. Therefore, we may construct a simplex intersecting the line $x_0 + tw$, $t \in \mathbb{R}$, by placing $n$ kissing balls on a plane $w^\perp$ perpendicular to $w$ and choosing a point from within each ball. Using simple geometrical arguments (cf. Apollonius's problem), it can be shown that these $n$ balls of radius $h$ are all contained within a larger ball of radius



(a) Interior simplex (b) Boundary simplex

FIG. 2. *Construction of simplices for the finite difference scheme on an interior point sufficiently far from the boundary (*2a*) and on a point near the boundary (*2b*).*

$C_n h$ (with $C_2 = 2$ and $C_3 = 1 + \frac{2}{\sqrt{3}}$). See Figure 1. Thus, a candidate simplex is guaranteed to exist within every ball of radius $C_n h$ with its center on the line $x_0 + tw$.

Let this larger ball be $\bar{B}(x_0 + (R - C_n h)w, C_n h)$. See Figure 2a. Simple trigonometric arguments show that this ball is contained within the cone $K$. Therefore, the cone $K$ contains the desired simplex $\mathcal{S}_p$.

Similar reasoning gives the existence of $\mathcal{S}_m$. Taken together, this allows for the construction of the schemes. □

LEMMA 3 (existence of interior scheme near boundary). *Take $x_0 \in \mathcal{G} \cap \Omega$ with* $\operatorname{dist}(x_0, \partial\Omega) < R$. *If the spatial resolution of $\mathcal{G}$ on the boundary is such that $C_n h_B \leq \delta \tan(\frac{d\theta}{2})$ and the angular resolution is small enough (dependent on the regularity of the boundary), then the scheme given by Definition* 1 *exists.*

*Proof.* We first will show $S_p$ exists; the existence of $\mathcal{S}_m$ follows analogously. With the cone $K$ defined as in the previous lemma, we must show that $\mathcal{G} \cap K$ contains points defining $\mathcal{S}_p$.

Suppose first that $\bar{B}(x_0, R) \cap K \subset \Omega$. Then the existence of $\mathcal{S}_p$ follows from Lemma 2.

Suppose instead that $\bar{B}(x_0, R) \cap K$ is not entirely contained within $\Omega$. If $d\theta$ is small enough, then a portion of the boundary is contained within $\bar{B}(x_0, R) \cap K$,

$$\|x_0 - y\| < R \quad \text{if } y \in \partial\Omega \cap K.$$

By construction, $\operatorname{dist}(x_0, \partial\Omega) \geq \delta$. Therefore, the diameter of this portion of the boundary is at least $\delta \tan(\frac{d\theta}{2}) \geq C_n h$. Using geometrical reasoning similar to that in the previous lemma (see Figure 2b), there must be $n$ points on the boundary defining the simplex $\mathcal{S}_p$. □

The previous two lemmas guarantee the first and second derivative schemes exist on the interior of the domain. The existence of the first derivative scheme on the boundary is, in general, not a simple exercise: existence depends on the regularity of the domain, the angle formed by $w$, and the boundary normal $n$, $h$, $h_B$, and $\delta$. For our purposes, we guarantee the existence of a scheme for the normal derivative with the following lemma.

Lemma 4 (existence of normal derivative scheme on the boundary). *Define the set $\Omega_\delta := \{x \in \Omega \mid \operatorname{dist}(x, \partial\Omega) \geq \delta\}$. Suppose $\Omega_\delta$ is such that for every $x \in \Omega_\delta$, $x \in \bar{B}(y, C_n h) \subset \Omega_\delta$ for some $y \in \Omega_\delta$. Suppose further that the minimum distance $\delta$ between interior points and boundary points is less than the* minimum *search radius $r$. Then the scheme $\mathcal{D}_n u(x_0)$ for the inward pointing normal derivative exists for all boundary points.*

*Proof.* Let $x_0$ be a boundary point. If $\delta < r$, then the search ball $\bar{B}(x_0 + (R - C_n h)n, C_n h)$ is contained entirely within $\Omega$. Thus, by the same arguments as in the proof of Lemma 2, the simplex $\mathcal{S}_p$ exists and has an angular resolution less than $d\theta$. This allows for the construction of (2) for the normal derivative. □

Combining these three lemmas guarantees existence of the schemes.

Theorem 1 (existence of schemes). *Suppose $\mathcal{G}$ is a point cloud in $\Omega$ with boundary resolution $C_n h_B \leq \delta \tan(\frac{d\theta}{2})$. With small enough $d\theta$, the first and second derivative schemes defined, respectively, by Definitions 1 and 2 exist for all interior points $x_0 \in \mathcal{G} \cap \Omega$. If in addition every $x \in \Omega_\delta$ lies within a ball $\bar{B}_{C_n h} \subset \Omega_\delta$ and $\delta < r$, the scheme $\mathcal{D}_n u(x_0)$ for the inward normal derivative exists for all boundary points $x_0 \in \mathcal{G} \cap \partial\Omega$.*

*Remark* 1 (boundary resolution). It is reasonable to expect that the minimum distance between interior points and the boundary is roughly equal to the spatial resolution, $\delta \approx h$. Since tan is nearly linear when $d\theta$ is small, Theorem 1 with optimal choice $d\theta \approx h^\alpha$ gives that $h_B = \mathcal{O}(h^{1+\alpha})$. The constant $\alpha$ is $\frac{1}{2}$ for regular grids and $\frac{1}{3}$ for point clouds; see Table 1.

**2.2. Consistency and accuracy.** We now derive bounds on the error of the schemes and show that the schemes are consistent with an appropriate choice of $d\theta$ in terms of $h$. First, recall the fact that the first term for the error of a linear interpolant is given by

$$u(x) - L(x) \approx \frac{1}{2} \sum \lambda_j(x)(x - x_j)^\intercal D^2 u(x_j)(x - x_j).$$

Therefore, the interpolation error at $x_0 + t_p w$ is

$$\begin{aligned} E[L_p] &:= u(x_0 + t_p w) - L_p(x_0 + t_p w) \\ &\approx \frac{1}{2} \sum_{i \in \mathcal{S}_p} \lambda_p^i \left(v_i^p - t_p w\right)^\intercal D^2 u(x_i) \left(v_i^p - t_p w\right) \\ &\leq \frac{1}{2} ||D^2 u||_\infty \sum_{i \in \mathcal{S}_p} \lambda_p^i ||v_i^p - t_p w||^2. \end{aligned}$$

The interpolation error at $x_0 - t_m w$ is bounded above in a similar fashion.

Lemma 5 (consistency of first derivative scheme). *The first derivative scheme of Definition 1 is consistent with a formal discretization error of $\mathcal{O}(h)$.*

*Proof.* The angular resolution error of the upwind first derivative scheme is

$$\begin{aligned} E[\mathcal{D}_w u, d\theta] &= \frac{E[L_p]}{t_p} \\ &\leq \frac{1}{2}\left\|D^2 u\right\|_\infty \sum_{i\in\mathcal{S}_p} \lambda_i^p \frac{\left\|v_i^p - t_p w\right\|^2}{t_p} \\ &\leq \frac{1}{2}\left\|D^2 u\right\|_\infty \frac{\max_{i,j\in\mathcal{S}_p}\left\|v_i^p - v_j^p\right\|^2}{\min_{k\in\mathcal{S}_p}\left\|v_k\right\|}. \qquad (3) \end{aligned}$$

By construction, the maximum distance between any two points in a simplex of the scheme is $2C_n h$, and so the numerator here is bounded above by $(2C_n h)^2$. Further, the minimum distance of a vector in the scheme is bounded below by the minimum search radius $r$. That is,

$$\begin{aligned} \min_{k\in\mathcal{S}_p,\mathcal{S}_m} \|v_k\| \geq r &= C_n h\left(-1 + \operatorname{cosec}\left(\frac{d\theta}{2}\right)\right) \\ &= \mathcal{O}\left(\frac{h}{d\theta}\right). \end{aligned}$$

With this in mind, (3) is bounded by

$$\begin{aligned} E[\mathcal{D}_w u, d\theta] &\leq \frac{1}{2}\left\|D^2 u\right\|_\infty \frac{(2C_n h)^2}{r} \\ &= \mathcal{O}(h d\theta). \end{aligned}$$

Fixing $d\theta$ constant as $h \to 0$ gives that the scheme is $\mathcal{O}(h)$. □

LEMMA 6 (consistency of second derivative schemes). *Using a nonsymmetric stencil, with the optimal choice* $d\theta = \left(\frac{h}{2}\right)^{\frac{1}{3}}$*, the second derivative scheme* $\mathcal{D}_{ww}u$ *of Definition* 2 *is consistent, with a formal accuracy of* $\mathcal{O}(h^{\frac{2}{3}})$*. Moreover, on a symmetric stencil, with the optimal choice* $d\theta = h^{\frac{1}{2}}$*,* $\mathcal{D}_{ww}u$ *is consistent, with a formal accuracy of* $\mathcal{O}(h)$*.*

*Proof.* The angular resolution error of the second derivative scheme is

$$\begin{aligned} E[\mathcal{D}_{ww} u, d\theta] &= 2\left(\frac{E[L_p]}{t_p^2 + t_p t_m} + \frac{E[L_m]}{t_m^2 + t_p t_m}\right) \\ &\leq \frac{1}{t_-^2}\Big(E[L_p] + E[L_m]\Big), \end{aligned}$$

where $t_- = \min\{t_p, t_m\}$. Arguing in a fashion similar to that in the first derivative,

$$\begin{aligned} E[\mathcal{D}_{ww} u, d\theta] &\leq ||D^2 u||_\infty \frac{\max_{S\in\mathcal{S}_p,\mathcal{S}_m}\max_{i,j\in S}||v_i - v_j||^2}{\min_{k\in\mathcal{S}_p,\mathcal{S}_m}||v_k||^2} \\ &\leq ||D^2 u||_\infty \frac{(2C_n h)^2}{r^2} \\ &= \mathcal{O}(d\theta^2), \end{aligned}$$

since $d\theta = \mathcal{O}(\frac{h}{r})$ when $d\theta$ is small.

The total error of the scheme is the sum of angular and spatial resolution errors (the spatial error arises from the finite difference series approximation and the angular error from the linear interpolation). For the second derivative, in the nonsymmetric case, the error of the scheme is

$$\begin{aligned} E[u_{ww}] &= \mathcal{O}(R + d\theta^2) \\ &= \mathcal{O}\left(\frac{h}{d\theta} + d\theta^2\right), \end{aligned}$$

because $R = \mathcal{O}(\frac{h}{d\theta})$ when $d\theta$ is small. In the symmetric case, the error is

$$\begin{aligned} E[\mathcal{D}_{ww} u] &= \mathcal{O}(R^2 + d\theta^2) \\ &= \mathcal{O}\left(\left(\frac{h}{d\theta}\right)^2 + d\theta^2\right). \end{aligned}$$

To ensure the scheme is consistent, $d\theta$ must be chosen in terms of $h$ such that the error of the scheme goes to zero as the point cloud is refined. In the nonsymmetric case, the best choice is $d\theta = \left(\frac{h}{2}\right)^{\frac{1}{3}}$, which gives a formal accuracy of $\mathcal{O}(h^{\frac{2}{3}})$. When the discretization is symmetric, the best choice of $d\theta$ is $\sqrt{h}$, and the scheme is formally $\mathcal{O}(h)$. □

*Remark* 2. To guarantee the accuracy of the first order scheme, $d\theta$ must remain constant as $h \to 0$. In contrast, for the second order scheme to converge as $h \to 0$, it must be that $d\theta \sim \left(\frac{h}{2}\right)^{\frac{1}{3}}$ (when the grid is not regular). Thus, for the remainder of the paper, when we speak of the angular resolution error, we mean the angular resolution error for the second derivative scheme. We assume that the angular resolution error for the first derivative scheme has been fixed to some reasonable constant, say $\frac{\pi}{4}$.

*Remark* 3. To ensure the existence of consistent schemes near the boundary, we require that the minimal distance between interior and boundary points is greater than the minimal search radius, $\delta \geq r$.

**2.3. Practical considerations.** We now outline a procedure for preprocessing the point cloud $\mathcal{G}$, which will greatly speed the construction of elliptic schemes. The algorithm takes a point cloud $x_i \in \mathcal{G}$, $i \in \mathcal{I}$ and returns a set $\mathcal{L}_i$ of candidate simplices for each point. Each simplex $\mathcal{S}_k \in \mathcal{L}_i$, $k = 1, \dots, m_i$, is contained within the annulus formed by the minimum and maximum search radii. Further, projecting $\mathcal{L}_i$ onto the sphere forms a covering of the sphere. Thus, all possible directions are available.

The pseudocode of the algorithm is given in Algorithm 1. Note that we assume the set of normalized neighbor points, denoted by $V$, is unique. If not, for each set of nonunique points, keep only the smallest direction satisfying the minimum search radius. This procedure is necessary, for example, on regular grids. We further assume that the triangulation is well-shaped in the following sense. For any two vertices $x_i$ and $x_j$ with $\|x_i - x_j\| < R$ in the triangulation, we assume that the graph distance between $x_i$ and $x_j$ is bounded above by $p(R)$.[1]

Now suppose the list of simplices

$$\mathcal{L}_i = \{S_k\}, \quad k = 1, \dots, m_i,$$

[1] For example, for a Delaunay triangulation in two dimensions, $p(R) = \frac{4\pi R}{3\sqrt{3}}$ [KG92].

**Algorithm 1:** Algorithm for preprocessing the point cloud.

```
Input  : A point cloud $x_i \in \mathcal{G}$ in $\mathbb{R}^n$, $i \in \mathcal{I}$, and resolution error $d\theta$
Output: A list of sets of simplices $\mathcal{L}_i$, $i \in \mathcal{I}$, where $\mathcal{L}_i = \{\mathcal{S}_1, \dots, \mathcal{S}_{m_i}\}$
$\mathcal{T} \leftarrow$ triangulation($\mathcal{G}$) ;                          // triangulation of $\mathcal{G}$
$A \leftarrow \mathrm{adj}(\mathcal{T})$ ;                                      // Adjacency matrix of $\mathcal{T}$
$h \leftarrow \sup_{x\in\Omega} \min_{y\in\mathcal{G}} \|x-y\|$ ;      // spatial resolution of point cloud
$R \leftarrow C_n h \left(1 + \operatorname{cosec}\left(\frac{d\theta}{2}\right)\right)$ ;                      // maximum search radius
$r \leftarrow C_n h \left(-1 + \operatorname{cosec}\left(\frac{d\theta}{2}\right)\right)$ ;                     // minimum search radius
$p \leftarrow \lceil p(R) \rceil$ ;                       // maximum neighbor graph distance
$P \leftarrow \sum_{k=1}^{p} A^k$ ;
foreach $i \in \mathcal{I}$ do
    $\mathcal{N} \leftarrow \{j \mid P_{ij} \neq 0, i \neq j, r \leq \|x_i - x_j\| \leq R\}$ ;      // Neighbor indices
    $V \leftarrow \left\{\frac{x_i - x_j}{\|x_i - x_j\|} \mid j \in \mathcal{N}\right\}$ ;        // assume elements of $V$ are unique
    $C \leftarrow$ Convex hull of $V$ ;
    $\mathcal{L}_i \leftarrow \emptyset$ ;
    foreach Facet $\mathcal{F}$ of $C$ do
        // $\mathcal{F}$ is a set of indices of the points in $V$
        $\mathcal{S} \leftarrow \{x_k \mid k = \mathcal{N}_j, j \in \mathcal{F}\}$ ;
        $\mathcal{L}_i = \mathcal{L}_i \cup \{\mathcal{S}\}$ ;
    end
end
return $\{\mathcal{L}_i\}$ , $i \in \mathcal{I}$
```

has been generated for a point $x_i$. Given a direction $w$, it is straightforward to choose $\mathcal{S}_p$ and $\mathcal{S}_m$ from $\mathcal{L}_i$. Define

$$V_k = \begin{bmatrix} v_1 & v_2 & \dots & v_n \end{bmatrix}, \qquad \text{with } v_k = x_j - x_i,\, j \in \mathcal{S}_k.$$

Then, by Farkas's lemma,

$$\mathcal{S}_p = \left\{\mathcal{S}_k \in \mathcal{L}_i \mid V_k^{-1} w \geq 0\right\}$$

and

$$\mathcal{S}_m = \left\{\mathcal{S}_k \in \mathcal{L}_i \mid V_k^{-1} w \leq 0\right\}.$$

If these sets are not singletons (when $w$ aligns with a grid direction), then choose one representative element.

*Remark* 4. The proofs of section 2 relied on choosing the maximal and minimal search radii to be $R, r = C_n h(\pm 1 + \operatorname{cosec}(\frac{d\theta}{2}))$, respectively. This choice makes the proofs relatively straightforward. However, it is possible to still guarantee existence and accuracy of the finite difference scheme with the narrower band of search radii $R, r = h(\pm 1 + C_n \operatorname{cosec}(\frac{d\theta}{2}))$. In practice, this set of search radii limits the appearance of "spikey" stencils. We have found that it is best to choose a set of simplices whose boundary has minimal surface area, thus limiting the amount of interpolation error.

## 3. Application: Eigenvalues of the Hessian.

It is relatively straightforward to employ $\mathcal{D}_{ww}u$ to find maximal and minimal eigenvalues of the Hessian about a point $x_i \in \mathcal{G}$. We will illustrate the procedure for the maximal eigenvalue, but the procedure is analogous for the minimal eigenvalue.

Define the finite difference operator $\Lambda_+^{h,d\theta} u(x_i) := \sup_{\|w\|=1} \mathcal{D}_{ww} u(x_i)$ as the approximation of the maximum eigenvalue of the Hessian.

Actually computing $\Lambda_+^{h,d\theta} u(x_i)$ reduces to an optimization problem. Define $K(\mathcal{S})$ as the cone generated by a set $S$. We say that two cones overlap if their intersection is nonempty. For each pair $\{\mathcal{S}_p, \mathcal{S}_m\}$ of overlapping antipodal simplices in $\mathcal{L}_i$ (with $K(\mathcal{S}_p) \cap K(-\mathcal{S}_m) \neq \emptyset$), one computes

$$\begin{aligned} P[\mathcal{S}_m, \mathcal{S}_p] = \underset{\lambda_p, \lambda_m}{\text{maximize}} \quad & 2\left[\frac{\sum_{i\in\mathcal{S}_p} \lambda_p^i u(x_i) - u(x_0)}{t_p^2 + t_p t_m} + \frac{\sum_{i\in\mathcal{S}_m} \lambda_m^i u(x_i) - u(x_0)}{t_m^2 + t_p t_m}\right] \\ \text{subject to} \quad & 0 \le \lambda_p, \lambda_m \le 1, \\ & 1^{\mathsf{T}} \lambda_p = 1, \\ & 1^{\mathsf{T}} \lambda_m = 1, \\ & t_p = ||V_p \lambda_p||, \\ & t_m = ||V_m \lambda_m||. \end{aligned}$$

The variables $t_p$ and $t_m$ are dummy variables. On a two-dimensional regular grid, this simplifies to an optimization problem over one variable, which can be solved analytically. (We note that in practice, we use an approximate method and do not solve this optimization problem directly. See Remark 5.)

To find the maximal eigenvalue, one takes the maximal value computed over all antipodal pairs:

$$\Lambda_+^{h,d\theta} u(x_i) = \max_{\substack{\mathcal{S}_m,\, \mathcal{S}_p \in \mathcal{L}_i \\ K(\mathcal{S}_m)\cap K(-\mathcal{S}_p)\neq\emptyset}} P[\mathcal{S}_m, \mathcal{S}_p]. \tag{4}$$

The error of the scheme is

$$\begin{aligned} E[\Lambda_+^{h,d\theta}] &= \left| \max_{\|v\|=1} v^{\mathsf{T}} D^2 u(x_i) v - \max_{\|w\|=1} \mathcal{D}_{ww} u(x_i) \right| \\ &\le \max_{\|w\|=1} w^{\mathsf{T}} D^2 u(x_i) w - \mathcal{D}_{ww} u(x_i) \\ &= \mathcal{O}(R + d\theta^2) \end{aligned}$$

on point clouds. As before, on a regular grid the error is $\mathcal{O}(R^2 + d\theta^2)$.

*Remark* 5. In cases other than on a regular grid in two dimensions, the optimization problem (4) is difficult to implement. In practice, as a compromise we instead compute finitely many directional derivative $\mathcal{D}_{w_i w_i} u$, $i = 1, \ldots, k$. Define the *effective* angular resolution through

$$\cos d\theta_e = \max_i \min_{j\neq i} \langle w_i, w_j \rangle.$$

Because the directional derivative may be taken off grid, one may choose sufficiently many directions $\{w_i\}$ such that $d\theta_e \le d\theta^2$. With this choice of directional derivatives, the maximal eigenvalue of the Hessian can be defined as

$$\Lambda_+^{h,d\theta_e} u(x_i) = \max_i \mathcal{D}_{w_i w_i} u(x_i).$$

A simple computation shows that $\Lambda_+^{h,d\theta_e}$ also has accuracy $\mathcal{O}(R + d\theta^2)$.

## 4. Solvers.

Before continuing with specific numerical examples, we first detail the numerical solver used. All solutions in section 5 were computed with a global semismooth Newton method. Without modification, the Newton method fails, because the Newton method is guaranteed to be only a local method. However, the Newton method achieves supralinear rates of convergence when the starting condition is close enough to the true solution.

Thus, to guarantee convergence, we use a global semismooth Newton method [FP07, Chapter 8]. Let $F^h[u]$ be a finite difference approximation of an elliptic operator $F[u]$. After each Newton step, we check for a sufficient decrease in the energy $\left\|F^h[u]\right\|^2$. If the Newton step does not decrease, the method switches to performing Euler steps, which is a guaranteed descent direction. We perform Euler steps for the same amount of CPU time as for one Newton step, which was first proposed in [Car17]. Because the Euler step is a guaranteed descent direction, the method is globally convergent [FP07].

## 5. Numerical examples.

Here we test our mesh-free finite difference method on two examples.[2] We demonstrate the convergence rates of the method and compare our method with that of [Fro18]. For each method, error is reported as the difference between the computed solution and the known analytic solution, measured in the maximum norm (which is the appropriate norm for measuring convergence to the viscosity solution [Obe06]).

### 5.1. Convex envelope.

Our first example is the convex envelope of a function $g(x)$ on a convex domain $\Omega$. The convex envelope has been well studied. In [Obe07], it was shown that the convex envelope solves the PDE

$$\begin{cases} \max\{u(x) - g(x), -\Lambda_- u(x)\} = 0, & x \in \Omega, \\ u(x) = g(x), & x \in \partial\Omega, \end{cases}$$

where $\Lambda_- u(x)$ is the minimal eigenvalue of the Hessian. A stable, monotone convergent finite difference scheme for computing the convex envelope was presented in [Obe08a].

In what follows, we take $g(x)$ to be the Euclidian distance to two points $p_1$ and $p_2$,

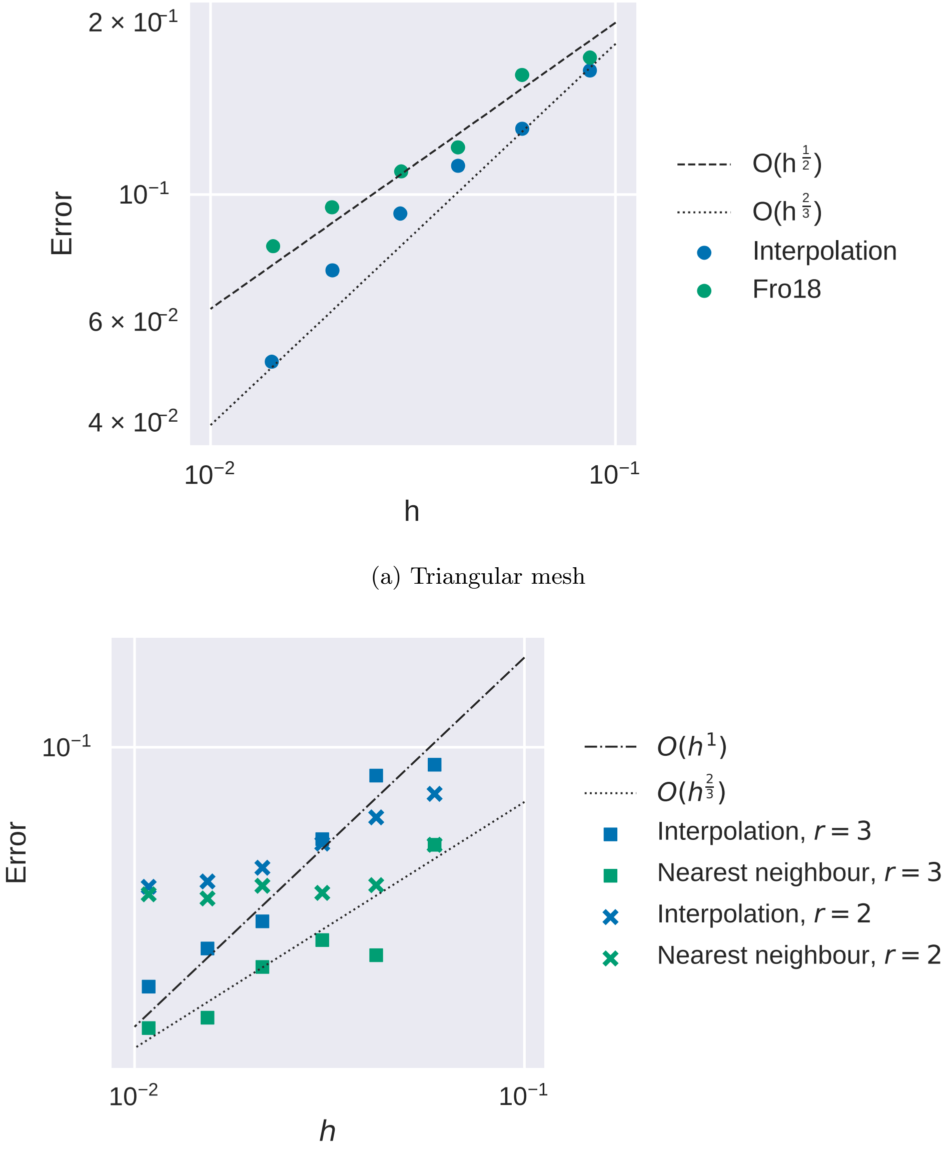
$$g(x) = \min_{i=1,2}\{\|x - p_i\|\},$$

or, in other words, a double cone.

We start by computing the solution on the square $[-1, 1]^2$, with $p_{1,2} = (\pm\frac{3}{7}, 0)$. We discretize $\Lambda_- u(x)$ using our symmetric linear interpolation finite difference scheme for eigenvalues of the Hessian, presented in section 3, and using the wide stencil method developed in [Obe08a]. We call the latter a nearest neighbor scheme. For both methods, we solved the equation using stencils with radii two and three. Figure 3b and Table 2 present convergence rates in the max norm. We can see that for stencil radius two, angular resolution error arises quickly as $h$ is decreased, and the error plateaus. However, with stencil radius three, we get a better handle on the convergence rate of the error. The standard wide stencil method achieves roughly $\mathcal{O}(h^{\frac{2}{3}})$, while the symmetric linear interpolation method achieves $\mathcal{O}(h)$, as expected.

Although the convergence rate of the linear interpolation method is better than the nearest neighbor method, for the values of $h$ we studied, the linear interpolation

[2] Our code, written in Python, is publicly available at https://github.com/cfinlay/pyellipticfd.



(a) Triangular mesh

(b) Regular grid

Fig. 3. *(*3a*): Convergence plot for the convex envelope on the unit disc with triangular mesh.* (3b)*: Convergence plot for the convex envelope on a regular grid over the square* $[-1, 1]^2$.

method has higher absolute error. This is because in order to guarantee convergence, the linear interpolation method must choose points greater than the minimum search radius, whereas the standard wide stencil finite difference scheme may choose its nearest neighbors. Thus, the spatial resolution error of the linear interpolation scheme is generally higher than the nearest neighbor scheme.

We are also interested in the error of the schemes as a function of the angular resolution. To this end, for fixed $h$, we compare the error of the schemes when the

Table 2
*Errors and convergence order for the convex envelope.*

| Triangular mesh, interpolation | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 8.6e−2 | 427 | 0.16 | – |
| 5.9e−2 | 785 | 0.13 | 0.61 |
| 4.0e−2 | 1452 | 0.11 | 0.41 |
| 2.9e−2 | 2713 | 0.09 | 0.58 |
| 2.0e−2 | 5101 | 0.07 | 0.59 |
| 1.4e−2 | 9674 | 0.05 | 1.07 |

| Triangular mesh, [Fro18] | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 8.6e−2 | 427 | 0.17 | – |
| 5.0e−2 | 810 | 0.16 | 0.18 |
| 4.1e−2 | 1533 | 0.12 | 0.80 |
| 3.0e−2 | 2908 | 0.11 | 0.30 |
| 2.0e−2 | 5526 | 0.09 | 0.37 |
| 1.4e−2 | 10542 | 0.08 | 0.47 |

| Regular grid, interpolation, $r = 2$ | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 5.9e−2 | 392 | 7.5e−2 | – |
| 4.1e−2 | 721 | 6.5e−2 | 0.43 |
| 3.0e−2 | 1288 | 5.5e−2 | 0.51 |
| 2.1e−2 | 2492 | 4.7e−2 | 0.43 |
| 1.5e−2 | 4616 | 4.3e−2 | 0.26 |
| 1.0e−2 | 9017 | 4.2e−2 | 0.09 |

| Regular grid, Nearest neighbor, $r = 2$ | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 5.9e−2 | 392 | 5.4e−2 | – |
| 4.2e−2 | 721 | 4.2e−2 | 0.73 |
| 3.0e−2 | 1288 | 4.0e−2 | 0.15 |
| 2.1e−2 | 2492 | 4.2e−2 | –0.13 |
| 1.5e−2 | 4616 | 3.9e−2 | 0.24 |
| 1.0e−2 | 9017 | 4.0e−2 | –0.07 |

| Regular grid, interpolation, $r = 3$ | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 5.9e−2 | 528 | 9.0e−2 | – |
| 4.2e−2 | 913 | 8.4e−2 | 0.20 |
| 3.0e−2 | 1552 | 5.6e−2 | 1.24 |
| 2.1e−2 | 2868 | 3.4e−2 | 1.45 |
| 1.5e−2 | 5136 | 2.9e−2 | 0.52 |
| 1.0e−2 | 9757 | 2.2e−2 | 0.68 |

| Regular grid, Nearest neighbor, $r = 3$ | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 5.9e−2 | 528 | 5.4e−2 | – |
| 4.2e−2 | 913 | 2.7e−2 | 2.0 |
| 3.0e−2 | 1552 | 3.0e−2 | –0.29 |
| 2.1e−2 | 2868 | 2.5e−2 | 0.47 |
| 1.5e−2 | 5136 | 1.9e−2 | 0.98 |
| 1.0e−2 | 9757 | 1.7e−2 | 0.18 |

grid has been rotated off axis. Our results are presented in Figure 4. The mean of the error of the linear interpolation scheme is higher than the nearest neighbor scheme, due to the fact that the linear interpolation scheme chooses points further from the stencil center. However, the variance of the error for the linear interpolation scheme is much less than that of the nearest neighbor scheme. That is, the linear interpolation scheme depends less on the angular resolution of the stencil relative to the rotation of the grid.

Finally, we compare the linear interpolation scheme with Froese's scheme on the unit disc, using an irregular triangulation of points. We generate the interior points using the triangulation software DistMesh [PS04] and augment the boundary with additional points to ensure a sufficient boundary resolution. Convergence rates are presented in Figure 3a and Table 2. We can see that the linear interpolation scheme achieves both the best rate of convergence and a better absolute error.

**5.2. Pucci equation.** Our next example is the Pucci equation,

$$\begin{cases} \alpha\Lambda_+ u(x) + \Lambda_- u(x) = 0, & x \in \Omega, \\ u(x) = g(x), & x \in \partial\Omega, \end{cases} \tag{5}$$

where $\alpha$ is a positive scalar, and $\Lambda_- u$ and $\Lambda_+ u$ are, respectively, the minimal and maximal eigenvalues of the Hessian. A convergent, monotone, and stable finite difference scheme for the Pucci equation was first developed in [Obe08b]. Following [DG05, Obe08b], we take

$$u(x, y) = -\rho^{1-\alpha}, \qquad \rho(x, y) = \sqrt{(x + 2)^2 + (y + 2)^2}.$$

We compute solutions on the unit disc and the square $[-1, 1]^2$.

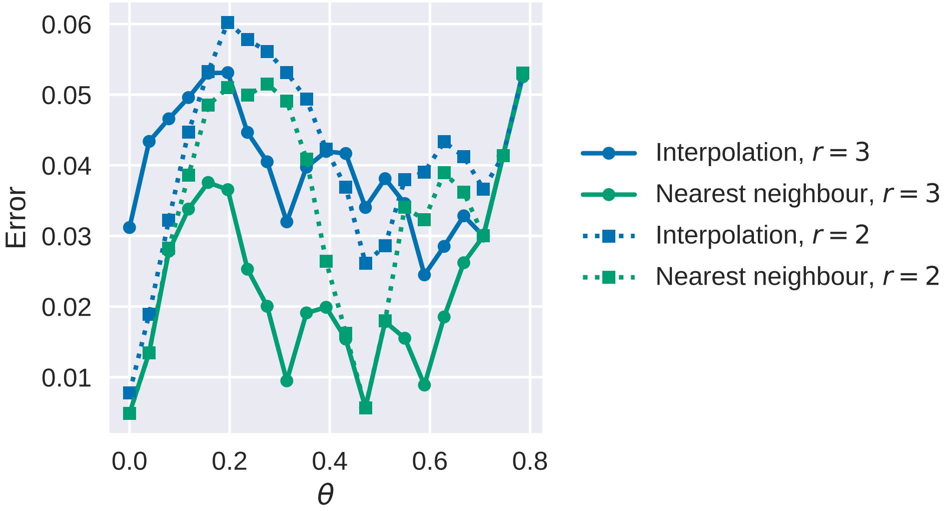


FIG. 4. *Error of the numerical solutions of the convex envelope PDE on a regular grid, as a function of rotation of the grid.*

We discretized the square using a regular grid and use either the nearest neighbor scheme or the symmetric finite difference interpolation scheme presented in section 3. We use stencils of radius two or three. Errors and rates of convergence on the grid are presented in Table 3 and in Figure 5b. Both methods achieve roughly the same convergence rate before angular resolution error dominates. The nearest neighbor scheme achieves a slightly better error rate.

As in the convex envelope example, we used DistMesh to triangulate the unit disc. Error and convergence rates are shown Table 3 and Figure 5a. Both methods achieve nearly $\mathcal{O}(h)$ convergence rate, which is better than predicted by our analysis. We hypothesize this is due to the fact that this example is smooth on the domain studied.

**5.2.1. Solver comparison.** Finally, we performed a comparison of the three solvers (semismooth Newton, Euler, and a combination of the two) in terms of CPU time for the Pucci equation on a regular grid. Results are presented in Table 4 and Figure 6.

As a function of number of grid points, the CPU time of Euler's method is roughly $\mathcal{O}(N^2)$ for both methods, interpolation and nearest neighbor. For the interpolation finite difference schemes, the CPU time of both semismooth Newton and the combination solver is nearly $\mathcal{O}(N)$: we calculated a log-log line of best fit and found the CPU time of the semismooth Newton and the combination solver to be about $\mathcal{O}(N^{1.2})$.

Of all solvers and finite difference methods, the nearest neighbor finite difference scheme with the combination solver achieves the best CPU time, followed by semismooth Newton. However, as a function of number of grid points, the CPU time is roughly $\mathcal{O}(N^{1.75})$. This rate is worse than the interpolation finite difference scheme, and so we expect on even larger grids that eventually the interpolation finite difference method would be faster with either semismooth Newton or the combination solver.

Table 3
*Errors and convergence order for the Pucci equation.*

| Triangular mesh, interpolation | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 8.6e−2 | 427 | 1.3e−3 | – |
| 5.0e−2 | 785 | 8.3e−4 | 1.16 |
| 4.1e−2 | 1452 | 5.0e−4 | 1.34 |
| 3.0e−2 | 2713 | 3.5e−4 | 1.09 |
| 2.0e−2 | 5101 | 2.5e−4 | 0.88 |
| 1.4e−2 | 9674 | 1.6e−4 | 1.29 |

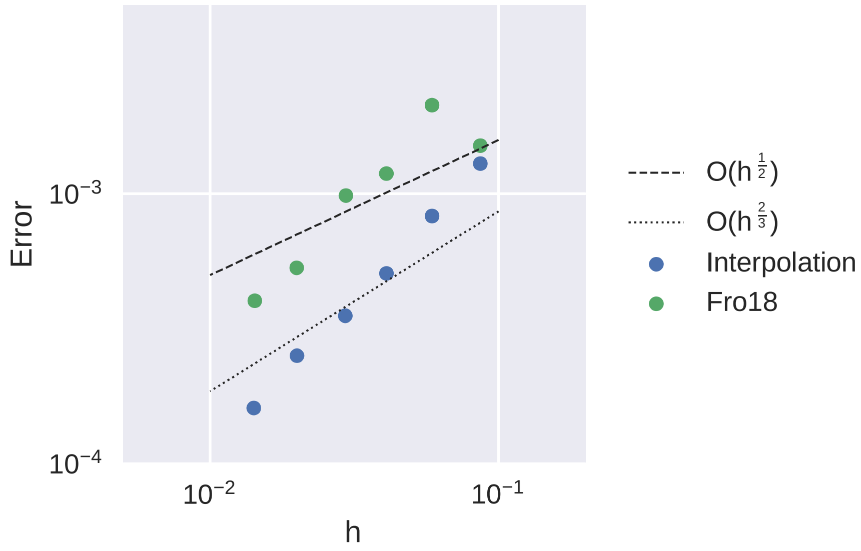
| Triangular mesh, [Fro18] | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 8.6e−2 | 427 | 1.5e−3 | – |
| 5.0e−2 | 810 | 2.1e−3 | –0.90 |
| 4.1e−2 | 1533 | 1.2e−3 | 1.60 |
| 3.0e−2 | 2908 | 9.9e−4 | 0.58 |
| 2.0e−2 | 5526 | 5.3e−4 | 1.57 |
| 1.4e−2 | 10542 | 4.0e−4 | 0.841 |

| Regular grid, interpolation, $r = 2$ | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 5.9e−2 | 392 | 9.4e−4 | – |
| 4.1e−2 | 721 | 7.0e−4 | 0.88 |
| 3.0e−2 | 1288 | 5.8e−4 | 0.58 |
| 2.1e−2 | 2492 | 5.1e−4 | 0.35 |
| 1.5e−2 | 4616 | 4.8e−4 | 0.19 |
| 1.0e−2 | 9017 | 4.6e−4 | 0.11 |

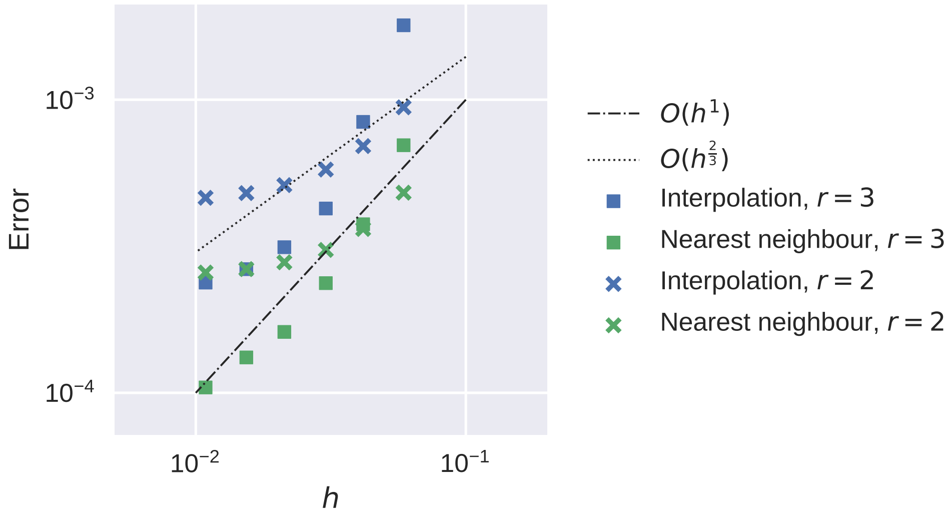
| Regular grid, Nearest neighbor, $r = 2$ | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 5.9e−2 | 392 | 4.8e−4 | – |
| 4.1e−2 | 721 | 3.6e−4 | 0.83 |
| 3.0e−2 | 1288 | 3.0e−4 | 0.52 |
| 2.1e−2 | 2492 | 2.8e−4 | 0.28 |
| 1.5e−2 | 4616 | 2.6e−4 | 0.16 |
| 1.0e−2 | 9017 | 2.6e−4 | 0.07 |

| Regular grid, interpolation, $r = 3$ | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 5.9e−2 | 528 | 1.0e−3 | – |
| 4.1e−2 | 913 | 8.4e−4 | 2.20 |
| 3.0e−2 | 1552 | 4.2e−4 | 2.14 |
| 2.1e−2 | 2868 | 3.1e−4 | 0.86 |
| 1.5e−2 | 5136 | 2.6e−4 | 0.53 |
| 1.0e−2 | 9757 | 2.3e−4 | 0.30 |

| Regular grid, Nearest neighbor, $r = 3$ | | | |
|---|---|---|---|
| $h$ | $N$ | Error | rate |
| 5.9e−2 | 528 | 7.0e−4 | – |
| 4.1e−2 | 913 | 3.8e−4 | 1.80 |
| 3.0e−2 | 1552 | 2.4e−4 | 1.45 |
| 2.1e−2 | 2868 | 1.6e−4 | 1.08 |
| 1.5e−2 | 5136 | 1.3e−4 | 0.62 |
| 1.0e−2 | 9757 | 1.0e−4 | 0.68 |



(a) Triangular mesh



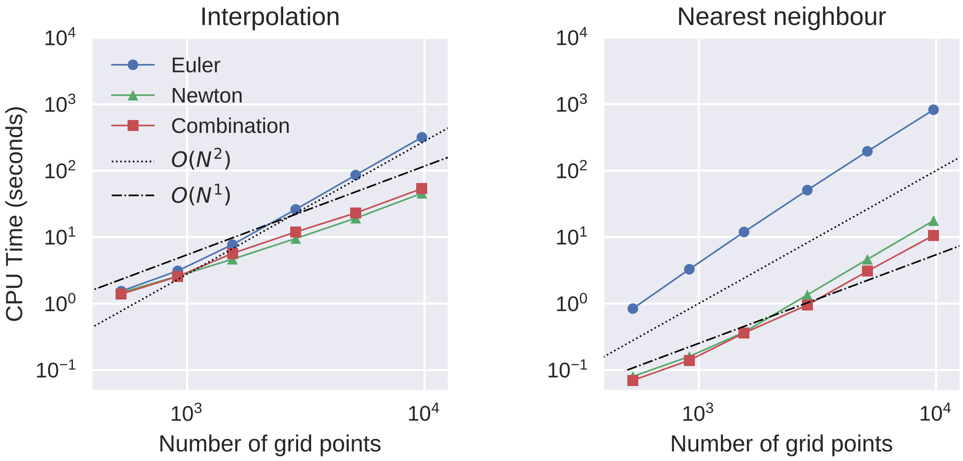
(b) Regular grid

FIG. 5. *(*5a*): Convergence plot for the Pucci equation on the unit disc with triangular mesh. (*5b*): Convergence plot for the Pucci equation on a regular grid over the square* $[-1, 1]^2$.

TABLE 4

*Comparison of wall clock time of solvers for the Pucci equation* (5) *in two dimensions on a regular grid. Time is reported in seconds. Results are for stencils of either radius* $r = 2$ *or* $r = 3$.

| | | | | | | |
|---|---|---|---|---|---|---|
| Interpolation, $r = 2$ | | | | | | |
| $N$ | 392 | 721 | 1288 | 2492 | 4616 | 9017 |
| Euler | 1.16 | 3.22 | 9.43 | 34.39 | 125.49 | 500.98 |
| Newton | 0.74 | 1.29 | 2.70 | 5.86 | 12.32 | 28.86 |
| Combination | 0.75 | 1.36 | 2.50 | 6.07 | 12.39 | 28.35 |
| Nearest neighbor, $r = 2$ | | | | | | |
| $N$ | 392 | 721 | 1288 | 2492 | 4616 | 9017 |
| Euler | 0.43 | 1.62 | 5.75 | 24.23 | 90.90 | 383.43 |
| Newton | 0.06 | 0.11 | 0.25 | 1.04 | 3.40 | 13.51 |
| Combination | 0.05 | 0.10 | 0.23 | 0.73 | 2.66 | 9.54 |
| Interpolation, $r = 3$ | | | | | | |
| $N$ | 528 | 913 | 1552 | 2868 | 5136 | 9757 |
| Euler | 1.54 | 3.11 | 7.73 | 26.17 | 86.05 | 317.98 |
| Newton | 1.47 | 2.59 | 4.66 | 9.54 | 19.29 | 45.68 |
| Combination | 1.39 | 2.56 | 5.70 | 11.97 | 23.12 | 53.77 |
| Nearest neighbor, $r = 3$ | | | | | | |
| $N$ | 528 | 913 | 1552 | 2868 | 5136 | 9757 |
| Euler | 0.84 | 3.28 | 11.93 | 50.90 | 195.14 | 823.99 |
| Newton | 0.08 | 0.16 | 0.37 | 1.35 | 4.63 | 17.66 |
| Combination | 0.07 | 0.14 | 0.36 | 0.95 | 3.07 | 10.61 |



(a) Triangular mesh

(b) Regular grid

FIG. 6. *CPU time taken to compute solution of the Pucci equation on a regular grid, with stencil width* $r = 3$, *for both methods.*

## REFERENCES

[BCM16] J.-D. Benamou, F. Collino, and J.-M. Mirebeau, *Monotone and consistent discretization of the Monge-Ampère operator*, Math. Comp., 85 (2016), pp. 2743–2775.

[BPR16] O. Bokanowski, A. Picarelli, and C. Reisinger, *High-order filtered schemes for time-dependent second order HJB equations*, ESAIM Math. Model. Numer. Anal., 52 (2018), pp. 69–97.

[BS91] G. Barles and P. E. Souganidis, *Convergence of approximation schemes for fully nonlinear second order equations*, Asymptotic Anal., 4 (1991), pp. 271–283.

[Car17] R. M. Carrington, *Speed Comparison of Solution Methods for the Obstacle Problem*, Master's thesis, McGill University, Montreal, QC, Canada, 2017.

[CIL92] M. G. Crandall, H. Ishii, and P.-L. Lions, *User's guide to viscosity solutions of second order partial differential equations*, Bull. Amer. Math. Soc. (N.S.), 27 (1992), pp. 1–67.

[CW17] Y. Chen and J. W. L. Wan, *Multigrid methods for convergent mixed finite difference scheme for Monge–Ampère equation*, Comput. Vis. Sci., (2017), pp. 1–15.

[CWL16] Y. Chen, J. W. L. Wan, and J. Lin, *Monotone mixed finite difference scheme for Monge–Ampère equation*, J. Sci. Comput., 76 (2018), pp. 1839–1867.

[DB08] G. Dahlquist and Å. Björck, *Numerical Methods in Scientific Computing*, Vol. 1, SIAM, Philadelphia, 2008, https://doi.org/10.1137/1.9780898717785.

[DG05] E. J. Dean and R. Glowinski, *On the numerical solution of a two-dimensional Pucci's equation with Dirichlet boundary conditions: A least-squares approach*, C. R. Math. Acad. Sci. Paris, 341 (2005), pp. 375–380.

[FFL+17] Z. Feng, B. D. Froese, R. Liang, D. Cheng, and Y. Wang, *Simplified freeform optics design for complicated laser beam shaping*, Appl. Opt., 56 (2017), pp. 9308–9314.

[FM14] J. Fehrenbach and J.-M. Mirebeau, *Sparse non-negative stencils for anisotropic diffusion*, J. Math. Imaging Vision, 49 (2014), pp. 123–147.

[FO13] B. D. Froese and A. M. Oberman, *Convergent filtered schemes for the Monge–Ampère partial differential equation*, SIAM J. Numer. Anal., 51 (2013), pp. 423–444, https://doi.org/10.1137/120875065.

[FP07] F. Facchinei and J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer-Verlag, New York, 2007.

[Fro18] B. D. Froese, *Meshfree finite difference approximations for functions of the eigenvalues of the Hessian*, Numer. Math., 138 (2018), pp. 75–99.

[FS17] B. D. Froese and T. Salvador, *Higher-Order Adaptive Finite Difference Methods for Fully Nonlinear Elliptic Equations*, preprint, https://arxiv.org/1706.07741, 2017.

[KG92] J. M. Keil and C. A. Gutwin, *Classes of graphs which approximate the complete Euclidean graph*, Discrete Comput. Geom., 7 (1992), pp. 13–28.

[Mir14a] J.-M. Mirebeau, *Anisotropic fast-marching on Cartesian grids using lattice basis reduction*, SIAM J. Numer. Anal., 52 (2014), pp. 1573–1599, https://doi.org/10.1137/120861667.

[Mir14b] J.-M. Mirebeau, *Minimal Stencils for Monotony or Causality Preserving Discretizations of Anisotropic PDEs*, hal-01086369v1, 2014.

[Mir16] J.-M. Mirebeau, *Adaptive, anisotropic and hierarchical cones of discrete convex functions*, Numer. Math., 132 (2016), pp. 807–853.

[NNZ19] R. Nochetto, D. Ntogkas, and W. Zhang, *Two-scale method for the Monge-Ampère equation: Convergence to the viscosity solution*, Math. Comp., 88 (2019), pp. 637–664.

[NSZ17] M. Neilan, A. J. Salgado, and W. Zhang, *Numerical analysis of strongly nonlinear PDEs*, Acta Numer., 26 (2017), pp. 137–303.

[Obe06] A. M. Oberman, *Convergent difference schemes for degenerate elliptic and parabolic equations: Hamilton–Jacobi equations and free boundary problems*, SIAM J. Numer. Anal., 44 (2006), pp. 879–895, https://doi.org/10.1137/S0036142903435235.

[Obe07] A. Oberman, *The convex envelope is the solution of a nonlinear obstacle problem*, Proc. Amer. Math. Soc., 135 (2007), pp. 1689–1694.

[Obe08a] A. M. Oberman, *Computing the convex envelope using a nonlinear partial differential equation*, Math. Models Methods Appl. Sci., 18 (2008), pp. 759–780.

[Obe08b] A. M. Oberman, *Wide stencil finite difference schemes for the elliptic Monge-Ampère equation and functions of the eigenvalues of the Hessian*, Discrete Contin. Dyn. Syst. Ser. B, 10 (2008), pp. 221–238.

[OS15] A. M. Oberman and T. Salvador, *Filtered schemes for Hamilton–Jacobi equations: A simple construction of convergent accurate difference schemes*, J. Comput. Phys., 284 (2015), pp. 367–388.

[OZ16] A. M. Oberman and I. Zwiers, *Adaptive finite difference methods for nonlinear elliptic and parabolic partial differential equations with free boundaries*, J. Sci. Comput., 68 (2016), pp. 231–251.

[PS04] P.-O. Persson and G. Strang, *A simple mesh generator in* Matlab, SIAM Rev., 46 (2004), pp. 329–345, https://doi.org/10.1137/S0036144503429121.