ON SOME APPLIED PROBLEMS USING NONLINEAR ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS

CHRISTOPHER FINLAY

Department of Mathematics and Statistics
FACULTY OF SCIENCE
McGill University, Montreal

JULY 2019

A thesis submitted to McGill University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

This dissertation studies several applied mathematical problems which broadly fall under the modelling framework of nonlinear elliptic partial differential equations (PDEs). Solutions to these problems are placed in the analytic setting of the theory of viscosity solutions; convergence of corresponding numerical solutions rely on the monotone schemes using the proof technique of Barles and Sougandidis.

The first two chapters study the problem of homogenization of nonlinear elliptic PDEs: find a macroscopic operator and corresponding solution that captures the behaviour of a rapidly varying microscopic operator, on broad scales. This is done through duality theory; we approximately solve an equivalent dual problem, which provides bound(s) on the true homogenized operator. Numerical experiments show that these approximate homogenized operators are quite accurate; in some cases error bounds are available using semi-concavity estimates.

The third chapter develops monotone finite difference schemes for nonlinear elliptic PDEs on point clouds. To date, most numerical methods for these equations have been on regular grids; motivated by the work of Froese this chapter extends monotone finite difference schemes to point clouds. The schemes rely on linear interpolation of neighbouring points in barycentric coordinates. Our schemes are of higher accuracy than previously available on both regular grids and point clouds. We prove consistency and stability of the schemes and provide several numerical examples in 2D.

The final chapter explores the problem of adversarial examples in image classification. These are small perturbations of an input image which cause a classifier to fail where a human would easily succeed. We address this problem using gradient regularization, which is inextricably linked to Tikhonov regularization in inverse problems and the p-Laplacian. We provide bounds on the minimum distance necessary to perturb an image adversarially, and show that gradient regularization improves these bounds. Moreover we implement gradient regularization in a scaleable fashion, using finite differences. This allows for quick training of adversarially robust models on very large datasets, which was previously intractable using prior methods.

ABRÉGÉ

Cette thèse porte sur l'étude de plusieurs problèmes de mathématiques appliquées, dont le cadre général est celui des équations aux dérivées partielles (EDP) elliptiques non-linéaires. La résolution de ces problèmes est obtenue dans le contexte de la théorie des solutions visqueuses; la convergence des solutions numériques correspondantes repose sur des schémas monotones, et utilise des techniques de preuves à la Barles et Sougandidis.

Les deux premiers chapitres s'intéressent à l'étude de l'homogénéisation d'EDP elliptiques non-linéaires: Trouver un opérateur macroscopique, et la solution correspondante qui capture à grande échelle le comportement de variations rapides d'un opérateur microscopique. Ceci est fait au travers du principe de dualité; la solution du problème dual équivalent est approximée, permettant de fournir une (des) borne(s) sur l'opérateur homogénéisé exacte. Les tests numériques montrent que l'approximation des opérateurs homogénéisés est précise; dans certains cas des bornes d'erreurs sont accessibles en utilisant des estimés semi-concaves.

Le troisième chapitre développe des schémas de différences finies monotones sur des nuages de points pour des EDP elliptiques non-linéaires. À ce jour, la plupart des méthodes numériques pour ce type d'équations sont définies sur des grilles régulières; motivé par les travaux de Froese, ce chapitre étend ces schémas de différences finies à des nuages de points. Ces schémas reposent sur une interpolation linéaire d'un voisinage de points dans un système de coordonnées barycentriques. Nos schémas ont un degré de précision plus élevé que ceux précédemment disponibles, tant sur des grilles régulières que sur des nuages de points. Des preuves de la consistences et de la stabilité des schémas sont données et plusieurs exemples en 2D sont présentés.

Le dernier chapitre explore le problème d'exemples antagonistes dans la classification d'images. Il s'agit de petites perturbations sur l'image d'entrée qui conduit à un échec du classifieur là où un humain réussit sans difficulté. Ce problème est traité au moyen de régularisation du gradient, et est intrinsèquement lié à la méthode de régularisation de Tikhonov des problèmes inverses et du p-Laplacien. Des bornes sur la distance minimale nécessaire pour perturber des images de manière adverse sont données et montrent que la régularisation du gradient permet de les améliorer. De plus, la méthode de régularisation du gradient est implémentée de manière adaptable à l'aide de différences finies. Ceci permet des entrainements rapides de modèles robustes à l'adversité sur un large ensemble

de données, ce qui était auparavant intractable au moyen de méthodes précédemment disponibles.

ACKNOWLEDGEMENTS

First and foremost, I'd like to thank my advisor Adam Oberman for his mentorship. I've come away from many a meeting energized by some beautiful derivations he blazed on the fly, or in awe of some new area of math I'd never encountered before. It's been a true pleasure working with him these past years.

I am grateful for funding from Mr Seymour Schulich through the Schulich Graduate Fellowship, and from Mr Taketo Murrata and Mr Alfred Murata through the Murata Family Fellowship. Your generous donations have given me the freedom to exclusively focus on research.

I'd like to thank Pratik Chaudhari for initiating me into the endlessly fasciniating world of machine learning and computer vision. My career has been irrevocably diverted towards the zeitgeist.

The McGill math & stats department has been a wonderfully supportive community. In particular I'd like to thank Tiago, Geoff, Bilal, and Tyler, my fellow applied math compatriots whom I befriended early on in the department. Tiago, for your incisive and scholarly criticism, every time I'd turn around with some hare-brained idea. Geoff, for your boundless enthusiasm, always ready with some dead-panned joke – and for tolerating my biannual attendence on the Wednseday night run club. Bilal for your friendship and afternoon coffee runs, always willing to listen to my academic gripes. Tyler, for keeping the Albertan work ethic in plain sight, and encouraging beer snobbery. I'd also like to thank all my officemates over the years, and the 10th floor denizens in general, for keeping the atmosphere in the concrete cheese-grater that is Burnside Hall relaxed and friendly.

Last, I must thank those close to my heart, for this thesis is a byproduct of your years of love and support. To my siblings and my parents, thank you. Your unconditional love is profound. To Clare, who set me on this path. I will remember, always. Finally to Emily, who has had infinite kindness and patience while I've worked on this PhD, thank you. I'm so happy to be on this journey together with you.

STATEMENT OF CONTRIBUTIONS

This thesis is comprised of four parts, each of which are based on an article co-authored by myself and are considered original scholarship. Briefly, the contributions of each chapter may be summarized as follows

Ch. 2 *Approximate homogenization of convex nonlinear elliptic PDEs* [FO18a] is a reproduction of an article of the same name, written by myself and Adam Oberman. It was submitted to Communications in Mathematical Sciences November 3, 2017, and accepted and published electronically in revised form May 28, 2018.

Ch. 3 *Approximate homogenization of fully nonlinear elliptic PDEs: estimates and numerical results for Pucci type equations* [FO18b] is a reproduction of an article of the same name, written by myself and Adam Oberman. It was submitted to the Journal of Scientific Computing on October 20, 2017; revised March 19, 2018; accepted May 5, 2018; and published online May 11, 2018.

Ch. 4 *Improved accuracy of monotone finite difference schemes on point clouds and regular grids* [FO18c] is a reproduction of a manuscript of the same name, written by myself and Adam Oberman. It was submitted to SIAM Journal on Scientific Computing (SISC) on July 13, 2018; revised April 4, 2019; and is currently under review.

Ch. 5 *Scaleable input gradient regularization for adversarial robustness* [FO19] is a reproduction of a manuscript of the same name, written by myself and Adam Oberman. It was submitted to the Thirty-third Conference on Neural Information Processing Systems (NeurIPS 2019) on May 23, 2019, and is currently under review.

The work of each chapter was done equally between myself and my advisor, Adam Oberman.

The code for the latter two chapters is available publically at `https://github.com/cfinlay`, and was written by myself.

You're path rises up to meet you. You must follow how it reveals itself to you. You can't think it through.

(C Patershuk)

CHAPTER 1

INTRODUCTION

This thesis is comprised of four parts, each of which is a self-contained manuscript and may be read indepently of the others. All chapters contain background material, a literature review, and concluding remarks; each stands on its own. Nevertheless, in this concise introduction we provide a high level overview of the material in this thesis, including general background material and key references, and thematically knit the contents into a cohesive whole.

BACKGROUND

This thesis is concerned with several applied problems which are placed within the context of elliptic partial differential equations (PDEs). Elliptic PDEs are operators for which a weak comparison principle holds: roughly, an operator F is said to be elliptic if $F[u] \leq F[v] \implies u \geq v$. The comparison principle is the main tool for proving uniqueness of solutions, by inducing an ordering on sub- and super-solutions (respectively those satisfying $F[u] \leq 0$ and $F[v] \geq 0$). In a heuristic way, elliptic PDEs behave analagously to Laplace's equation, a basic example of an elliptic operator.

The correct framework for studying solutions of elliptic PDEs is that of *viscoscity solutions*, introduced by Crandall, Lions and Evans in a series of papers [Eva80, CL83, CEL84]. For a definitive overview, see [CIL92]. Because they are typically not differentiable everywhere, viscosity solutions do not satisfy elliptic PDEs in a classical sense. However, viscosity solutions are a type of weak solution, in that the requirement of differentiability is passed to C^2 test functions as follows. Roughly, an upper semi-continuous function u is said to be a viscosity sub-solution at a point x if for every C^2 test function ϕ such that $u - \phi$ attains a maximum at x, with $u(x) = \phi(x)$, the inequality $F[\phi] \leq 0$ holds. Pictorially, ϕ grazes u from above at x. Similarly, super-solutions may be defined. A viscosity solution is then both a sub- and super-solution.

The great power of the viscosity solution framework is that it allows for solutions of a truly vast number of nonlinear elliptic PDEs arising in physics, engineering and elsewhere for which classical solutions are not available. For example, the Monge-Ampère equation (arising in optimal transport [Vil08]); the p-Laplacian (semi-supervised learning [Cal17], image denoising [RO94]); Hamilton-Jacobi-Bellman equations (stochastic optimal control

[FS06]); Isaacs equations (stochastic differentiable games [BEJ84]); and affine curvature (edge detection, image analysis [Sap06]) are all described by nonlinear elliptic PDEs for which the theory of viscosity solutions applies.

Approximate homogenization of nonlinear elliptic PDEs

Chapters 2 and 3 are concerned with the approximate homogenization of non-linear elliptic PDEs. Homogenization is the problem of replacing an operator $F^\varepsilon[u]$ that depends on a microscopic scale ε, with another macroscopic operator $F[u]$. The operator $F[u]$ does not depend on the microscopic scale ε. It is hoped that solutions of $F[u] = 0$ agree with solutions $F^\varepsilon[u^\varepsilon] = 0$ closely, with error on the order of ε. In other words, homogenization seeks to find a limiting PDE as $\varepsilon \to 0$, with solutions u^ε converging uniformly to u.

In [Eva89], Evans showed that for nonlinear elliptic PDEs, the viscosity solution framework gives a means for tackling this problem, by passing the limit problem onto smooth perturbed test functions. At the end of the day, this amounts to solving the *cell problem*. For a second-order operator, for a symmetric $Q \in \mathcal{S}^d$, the cell problem is to find a unique value $\bar{F}$ and a periodic viscosity solution u^ε which satisfy

$$F^\varepsilon(Q + D^2 u^\varepsilon(y)) = \bar{F} \tag{1.1}$$

The value $\bar{F}$ implicitly depends on Q, in fact this is the homogenized operator.

There is a large array of theoretical homogenization literature, including homogenization in stochastic environments, and rates of convergence (see for example [AKM19] for recent work on divergence-form operators). For a general review, see [ES08]. However, there are few works on analytic representations of homogenized operators. Chapters 2 and 3 aim to address this gap, for some families of periodic operators.

In Chapter 2 we obtain analytic representations of the homogenization of fully nonlinear, convex uniformly elliptic PDEs. Rather than solving the cell problem directly, convexity of the operator allows for a solution of the cell problem to be found using duality arguments. In particular, the cell problem has an associated dual problem, with an associated solution, called the *optimal invariant measure*. It is often easier to solve the dual problem than the cell problem. Notably, non-optimal solutions to the dual problem provide a lower bound on the true solution of the cell problem. This is the approach taken in Chapter 2: we approximately solve the dual problem to obtain (approximate) analytic representations of the homogenized operator, for several concrete examples. We show numerically (through monotone finite difference numerical schemes, see below) that these approximate homogenizations are 'good enough' in many regimes. In fact the error

of these approximate homogenizations is only large near areas of high curvature in the operator.

In Chapter 3 we tackle approximate homogenization of Pucci-type operators. These are operators that are sums of the minimal and maximal eigenvalues of the Hessian, and are non-convex in general. Pucci-type operators are important objects of study in their own right, for they provide bounds on viscosity solutions of uniformly elliptic equations in non-divergence form [CC95]. Our approximate homogenization is obtained by first linearizing the operator, which leads to a non-divergence form homogenization problem which approximately solves the original. We homogenize this linearized operator as in Chapter 2, using the optimal invariant measure. We obtain error bounds of the approximate homogenization using semi-concavity estimates on the original operator. Numerical results show that the approximate homogenization is remarkably accurate away from areas where the original operator is non-differentiable.

Monotone finite difference schemes on point clouds

In practice, elliptic PDEs must be solved with numerical methods. This is the focus of Chapter 4. Provably convergent numerical schemes are available via the seminal work of Barles and Sougandidis [BS91]. Briefly, schemes converge to the unique viscosity solution of the underlying elliptic PDE if they (i) respect the ordering induced by the comparison principle of the underlying operator (said to be monotone), (ii) are stable, and (iii) are consistent. However, given an elliptic PDE, the construction of a monotone, stable and convergent scheme is not at all obvious *a priori*. In [Obe06], a framework for the construction of these schemes was developed using so-called *elliptic schemes*, and are the natural framework for construction of monotone finite difference schemes. This led to the development of wide-stencil schemes on regular grids, which are needed for elliptic schemes of second-order elliptic PDEs, as in the Monge-Ampère equation (arising from Optimal Transport) or Pucci-type operators [Obe08b].

Although wide-stencil elliptic schemes are provably convergent, they often suffer from low accuracy, because they depend both on the spatial resolution of the discretization, and the width of the stencil. In [FO13] this issue was addressed using filtered schemes, which combine elliptic schemes with more accurate (but are potentially unstable) methods, while still being provably convergent. However because filtered schemes are built on elliptic schemes, it is still desirable to search for more accurate elliptic schemes. This is the purpose of Chapter 4, where we develop more accurate wide-stencil elliptic schemes than previously available.

Moreover, there has been much recent interest in extending elliptic schemes to mesh-free point clouds, for example in freeform optic design for laser beam [FFL+17]. In [Fro18], Froese proposed a mesh-free elliptic scheme for point clouds in two dimensions. Motivated by this work, in Chapter 4 we extend the method presented therein to point clouds in n-dimensons. The method is compared to that of [Fro18], and is shown to have higher asymptotic accuracy.

Gradient regularization for adversarial robustness

Although not obvious at first, there is a deep connection between elliptic PDEs and Chapter 5, which is concerned with the construction of *adversarially robust* models in machine learning regression and classification problems. In both regression and classification problems, the modeling task is to learn a map $u : \mathcal{X} \to \mathcal{Y}$ that closely matches a label function f, provided some distribution ρ on $\mathcal{X}$. Typically $\mathcal{X}$ is a closed and bounded subset of $\mathbb{R}^d$. The learned function u is restricted to some smaller function class $\mathcal{F}$: for example, a simple choice is the space of affine functions; whereas kernel based methods represent u approximately as a finite sum of basis elements (feature maps in ML parlance) of a Hilbert space (see for example [MRT18, Ch. 6]). Also in vogue, $\mathcal{F}$ may be represented via a neural network architecture.

Irrespective of the function class, u is found by minimizing a functional. In regression, the squared error is error used: u solves

$$\min_{u \in \mathcal{F}} \int_{\mathcal{X}} (u - f)^2 \, \mathrm{d}\rho \tag{1.2}$$

In classification, the Kullback-Leibler (KL) divergence is minimized instead. Of course, in real-world applications f is not known everywhere (for otherwise we would simply forgo this entire process and use f instead), but merely on a subset of m samples $\{(x_i, f_i)\}$, $i = 1, \ldots, m$, where the x_i's are drawn randomly according to the distribution ρ. To find u, the empirical risk is minimized

$$\min_{u \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^{m} (u(x_i) - f_i)^2 \tag{1.3}$$

Unfortunately, when $\mathcal{F}$ is broad enough, naively minimizing the empirical risk leads to a host of problems. Foremost among these is a failure to generalize, also known as over-fitting. Given a broad enough function class, the learned function may perfectly minimize the empirical risk perfectly, yet have large error over the entire distribution ρ.

Equally troubling is the occurrence of *adversarial examples*, first observed in neural

networks by [BCM+13] and [SZS+13] in the context of computer vision classification. Even supposing a learned function u generalizes well, it is still possible to imperceptibly perturb real-world images (to the human eye) so that predictions made by u are wildly incorrect, when a human would have no trouble at all labeling the image correctly. Hence without further action, neural networks are not adversarially robust.

Many solutions aimed at addressing this problem have been proposed, yet prior to our work none to-date have scaled well to very large datasets. In Chapter 5, we propose a fast and scaleable implementation of gradient regularization for adversarial robustness in neural networks. Ideally we would regularize (1.2) with a gradient penalty term, similar to

$$\min_{u \in \mathcal{F}} \int_{\mathcal{X}} (u - f)^2 + |\nabla u|^2 \, \mathrm{d}\rho, \tag{1.4}$$

which would penalize solutions for being susceptible to adversarial perturbations. In many fields this is known as a form of Tikhonov regularization [TA77] and has a long history of study, especially in inverse problems.

However Chapter 5 is concerned primarily with classification, where the function u returns a classification probability, and so it is more convenient to consider a slightly modified version of (1.4). For the quadratic loss, we suggest penalizing large gradients of the loss via

$$\min_{u \in \mathcal{F}} \int_{\mathcal{X}} (u - f)^2 + \lambda |\nabla u - \nabla f|^2 \, \mathrm{d}\rho \tag{1.5}$$

where λ can be thought of as a Lagrange multiplier controlling the regularization strength. Chapter 5 discusses a similar penalty, for large gradients of the KL loss.

The variational problem (1.4) is intimately tied to an elliptic partial differential equation. The Euler-Lagrange equation associated with (1.4) is

$$u - \frac{1}{\rho} \operatorname{div}(\rho \nabla u) = f \tag{1.6}$$

which is slightly modified when considering (1.5). A similar elliptic PDE arises in semi-supervised learning, in which labels f_i are only present on a subset of the sampled data. In this scenario, the variational problem is to minimize

$$\min_{u \in \mathcal{F}} \int_{\mathcal{X}} |\nabla u|^p \, \mathrm{d}\rho \quad \text{subject to } u \equiv f \text{ on available data.} \tag{1.7}$$

Minimizers of (1.7) satisfy the p-Laplacian, $-\operatorname{div}(\rho |\nabla u|^{p-2} \nabla u) = 0$. For a discussion of the connections between semi-supervised learning and the p-Laplacian, see [Cal17, Cal19], where it was shown that in the limit of infinite unlabeled data but finite labeled data (1.7) is well posed only when $p > d$. In a certain sense, the adversarial robustness problem

(1.4) can be viewed as a soft relaxation of (1.7), in which the hard constraint has been incorporated as a penalty term via the squared error $(u - f)^2$.

In low dimension, the p-Laplacian and (1.6) may be solved numerically with wide-stencil elliptic finite difference schemes, discussed in Chapter 4. For example, in Chapter 4 we solve the ∞-Laplacian using the higher accuracy wide-stencil method developed therein, with the p-Laplacian finite difference schemes proposed in [Obe05, Obe13].

Though in practice PDE based methods are intractable in the high dimensional setting of machine learning, we nevertheless draw inspiration from the numerical analysis literature, and use finite differences to approximate the gradient regularization term in (1.4), opting to solve (1.4) directly. This allows us to avoid 'double backpropagation' (applying automatic differentiation twice), and is to our knowledge the first method for adversarial robustness which scales efficiently to very large datasets like ImageNet-1k [DDS+09]. We also provide theoretical lower bounds on the minimum distance necessary to adversarially perturb an image, and empirically show that gradient regularization improves these bounds.

CHAPTER 2

APPROXIMATE HOMOGENIZATION OF CONVEX NONLINEAR ELLIPTIC PDES

Abstract

We approximate the homogenization of fully nonlinear, convex, uniformly elliptic Partial Differential Equations in the periodic setting, using a variational formula for the optimal invariant measure, which may be derived via Legendre-Fenchel duality. The variational formula expresses $\overline{H}(Q)$ as an average of the operator against the optimal invariant measure, generalizing the linear case. Several nontrivial analytic formulas for $\overline{H}(Q)$ are obtained. These formulas are compared to numerical simulations, using both PDE and variational methods. We also perform a numerical study of convergence rates for homogenization in the periodic and random setting and compare these to theoretical results.

2.1 INTRODUCTION

We consider homogenization of the periodic, convex, uniformly elliptic Hamilton-Jacobi-Bellman operator

$$H(Q,y) = \sup_{\alpha\in\mathcal{A}} L_\alpha(Q,y) = \sup_{\alpha\in\mathcal{A}} \left\{-A(y,\alpha) : Q - h(y,\alpha)\right\}. \tag{2.1}$$

Note that $H(Q,y)$ is convex in Q. Let $\mathcal{A}$ be a convex and closed control set, and let $A : \mathbb{T}^d \times \mathcal{A} \to \mathcal{S}^d$, where $\mathcal{S}^d$ is the space of $d \times d$ symmetric matrices, and $\mathbb{T}^d$ is the d-dimensional torus. Let $h : \mathbb{T}^d \times \mathcal{A} \to \mathbb{R}$ be continuous and convex in α. We assume that A is uniformly elliptic, with $0 \ll \lambda I \ll A \ll \Lambda I$. If $\mathcal{A}$ is not compact, then we also require h to be superlinear in α, that is

$$\lim_{|\alpha|\to\infty} \frac{h(y,\alpha)}{|\alpha|} = \infty, \quad \forall y \in \mathbb{T}^d.$$

We will make use of the following result, stated in Corollary 2.1.4, and which follows from Theorem 3.2.2 below. Consider an admissible control $\alpha(y)$, $\alpha : \mathbb{T}^d \mapsto \mathcal{A}$, and suppress writing the dependence on y explicitly. Let L_α be the corresponding linear operator,

$$L_{\alpha(y)}(Q,y) = -A\left(y,\alpha(y)\right) : Q - h\left(y,\alpha(y)\right). \tag{2.2}$$

Let $\overline{L_\alpha}(Q)$ be the homogenized linear operator. Then

$$\overline{L_\alpha}(Q) \leq \overline{H}(Q). \tag{2.3}$$

Equality holds when the control corresponds to linearizing $H(Q, y)$ in Q about the corresponding solution u^Q of the cell problem (see Definition 3.4 below, or [Eva89, Eva92]). That is, when $a(y)$ satisfies

$$L_{\alpha(y)}(Q + D^2u^Q, y) = H(Q + D^2u^Q(y), y) = \overline{H}(Q).$$

Equivalently, equality holds when the support of the optimal invariant measure (see Definition 2.1.2) concentrates on $\alpha(y)$.

In Section 2.2 we consider four example problems. One example is the maximum of two linear operators. In this case, we obtain a formula for $\overline{H}(Q)$, which is new (as far as we know). The second example is one dimensional, but with a quadratic nonlinearity. In this case, by considering constant controls, we find a lower bound for $\overline{H}(Q)$ which is numerically verified to be sharp.

The third example is a two dimensional Pucci operator on stripes. In [FO18b] we homogenized Pucci operators, mainly with checkerboard coefficients. There we did not require convexity of the operator. We obtained accurate results for values of Q away from the singularities of the operators by simply linearizing the operator about Q. However, for stripes coefficients, the linearization about Q is not accurate. Here, we linearize about a control, and find the optimal constant control which corresponds to a control direction which depends on both the eigenvectors of Q and the orientation of the stripes. When compared numerically to $\overline{H}(Q)$, this control gives very accurate results, away from the singularities. Near the singularities, there is still a small nonzero error. In [FO18b] we also established upper bounds for the linear homogenization error. These estimates included a term which decreased with the distance to the singular set of the operators. Similar results apply here as well.

The last example is a separable operator in one dimension, $H(Q, y) = a(y)H_0(Q)$. We obtain the exact representation $\overline{H}(Q) = \mathrm{HM}(a)H_0(Q)$, where $\mathrm{HM}(a)$ is the harmonic mean of a.

We compared our estimates for $\overline{H}(Q)$ to numerical results. We computed $\overline{H}(Q)$ numerically using two methods: by solving the PDE for the cell problem, and by using linear programming to solve for the invariant measure. We also considered the case of random coefficients, and we found that very similar formulas for $\overline{L_\alpha}(Q)$ hold in the random setting.

We also computed rates of convergence for $\overline{H}(Q)$. In the periodic case, we obtained

second order convergence rates, $\mathcal{O}(\varepsilon^2)$, in one dimension. In the random setting we obtained a convergence rate of $\mathcal{O}(\varepsilon^{1/2})$, again in one dimension. These are consistent with the theoretical results we mention below.

2.1.1 *Related work*

We know of few analytical solutions $\overline{H}(Q)$, other than the formula for a first order Hamiltonian in one dimension which can be found in the early paper [LPV87]. In [FO18b] we obtained an approximation of $\overline{H}(Q)$ in terms of its linearization about Q. The error of this approximation depends only on a generalization of the semiconcavity constant of the operator. However, in [FO18b] we found examples of Pucci type operators where the numerically compute value of $\overline{H}(Q)$ is very close to the approximation, for values of Q away from the corners of the operator.

For a general reference on theoretical and numerical homogenization in this context, we refer to the review paper [ES08].

A numerical method which uses the $\inf\sup$ formula for the first order case was developed in [GO04]. In [OTV09] we studied homogenization of convex (first order) Hamilton-Jacobi equations; some exact formulas in the periodic setting can be found there. Recently [CC16] studied numerical homogenization of mainly first order equations, along with one dimensional second order equations. In [CG08] the problem of homogenization of a Pucci type equation with checkerboard coefficients was studied. In that case, our results are close to, but different from theirs, see [FO18b].

In the random setting, the first qualitative homogenization results for fully nonlinear uniformly elliptic operators were obtained in [CSW05], followed by [CS10] which established a logarithmic estimate for convergence rates in strongly mixing environments. Algebraic convergence estimates were established in [AS14], where it was shown that in a uniformly mixing environment,

$$\mathbb{P}\left[\|u^\varepsilon - \bar{u}\|_\infty \geq C\varepsilon^\beta\right] \leq C\varepsilon^\beta,$$

where C and β are constants that do not depend on ε. In the periodic case [CM09], proved that the order of convergence for the cell problem is $\mathcal{O}(\varepsilon^2)$, when the HJB operator does not depend on first order terms or the macroscopic scale.

2.1.2 *Background Theory*

In the periodic uniformly convex setting, the homogenized operator can be obtained by solving the cell problem.

Definition 2.1.1 (Cell problem). *Given H as in (2.1), for each $Q \in \mathcal{S}^d$, there is a unique value $\overline{H}(Q)$ and a periodic function $u^Q(y)$ which is a viscosity solution of the cell problem*

$$H(Q + D^2 u^Q(y), y) = \overline{H}(Q). \tag{2.4}$$

Because the operator H is uniformly elliptic, one can show that both the value $\overline{H}(Q)$ and the solution u^Q exist and are unique, and that $u^Q \in C^2(\mathbb{T}^d)$ [Eva89].

In the linear case, we may use the Fredholm Alternative to find the invariant measure, and the homogenized operator is then obtained by averaging against the invariant measure, see for example [BLP11] and [FO09]. That is, under an integral compatibility condition, there is a unique invariant probability measure, ρ, which solves $D^2 : (A(y)\rho) = 0$, and the homogenized PDE operator is $\bar{L}(Q) = \bar{A} : Q + \bar{h}$ where $\bar{A} = \int_{\mathbb{T}^d} A \, \mathrm{d}\rho$ and $\bar{h} = \int_{\mathbb{T}^d} h \, \mathrm{d}\rho$.

In the nonlinear case, the homogenized operator may still be found by averaging against the optimal invariant measure (see [Gom05] or [IMT16]).

Definition 2.1.2 (Optimal invariant measure). *Let $\Pr(\mathbb{T}^d \times \mathcal{A})$ denote the space of Borel probability measures on $\mathbb{T}^d \times \mathcal{A}$. For $\rho \in \Pr(\mathbb{T}^d \times \mathcal{A})$, we say ρ is an* invariant measure *if $L_0^* \rho(y, \alpha) = 0$ hold in the weak sense, by which we mean*

$$\int_{\mathbb{T}^d \times \mathcal{A}} A(y, \alpha) : D^2 \varphi(y) \, \mathrm{d}\rho(y, \alpha) = 0, \quad \forall \varphi \in C^2(\mathbb{T}^d). \tag{2.5}$$

Define

$$\bar{H}_{LP}(Q) = \sup_{\rho \in \Pr(\mathbb{T}^d \times \mathcal{A})} \left\{ \int L_\alpha(Q, y) \, \mathrm{d}\rho(y, \alpha) \;\middle|\; L_0^* \rho = 0 \right\}, \tag{2.6}$$

The fact that (2.6) and (3.4) give the same value is established in Theorem 2.1.3, a proof of which may be found in [Gom05] or [IMT16]. The result follows from duality and convex analysis, in particular the theorem of Fenchel-Rockafeller.

Theorem 2.1.3. *Let $\overline{H}(Q)$ be defined by (3.4) and $H_{LP}(Q)$ be defined by (2.6). Then*

$$\overline{H}(Q) = \bar{H}_{LP}(Q).$$

Remark 2.1. The formula above expresses an optimal invariant measure as a maximizer of the functional in (2.6), and the homogenized operator as the average of $L_\alpha(Q, y)$ against an optimal invariant measure.

Note that while the optimal invariant measure depends on Q, the set of invariant measures does not. This allows us to sometimes find $\bar{H}(Q)$ for all Q once the invariant measures are determined. While $\bar{H}(Q)$ does not depend on how we represent $H(Q, y)$ in

(2.1), the set of invariant measure does. So a more concise representation of the operator can lead to a smaller set of invariant measures.

Corollary 2.1.4. *Let $\overline{H}(Q)$ defined by (3.4) and let $\overline{L_\alpha}(Q)$ be the homogenization of (2.2). Then $\overline{L_\alpha}(Q) \leq \overline{H}(Q)$.*

Proof. $\overline{L_\alpha}(Q)$ is obtained by averaging $L_\alpha(Q, y)$ against ρ_α, the invariant measure with support on the graph $\alpha(y)$ and satisfying $D^2 \left(A(y, \alpha(y)\rho_\alpha\right) = 0$. Inequality follows from (2.6) and Theorem 3.2.2. □

2.2 Estimates from Linearization

In this section we apply (2.3), by considering specific operators where we can obtain analytical values for the homogenized linear operator $\overline{L_\alpha}(Q)$. The general procedure is to put the operator in HJB form (2.1), solve for an invariant measure satisfying (2.5) for a fixed control at the linear operator L_α, homogenize the linear operator, and then maximize over all controls.

2.2.1 *Pucci type operators on stripes*

Consider the following Pucci type operator.

Definition 2.2.1. *Let $y \in \mathbb{T}^2$ and $Q \in \mathcal{S}^2$. Write $\lambda_{max}(Q), \lambda_{min}(Q)$ for the maximum and minimum eigenvalues of Q, respectively. Given $b(y) \geq 0$ and $a(y) > 0$, define the convex Pucci type operator*

$$F^{a,b}(Q, y) = a(y) \operatorname{Tr} Q + b(y)\lambda^+_{max}(Q) \tag{2.7}$$

where $t^+ := \max\{t, 0\}$

Remark 2.2. The operator (3.19) may be recast into the form of (2.1) by setting

$$L_{\mathbf{v}}(Q, y) = A(y, \mathbf{v}) : Q$$

where $A(y, \mathbf{v}) = a(y)I + b(y)\mathbf{v} \otimes \mathbf{v}$ and $\mathcal{A} = \{|\mathbf{v}| \leq 1\}$. Note that $h(y, \mathbf{v}) \equiv 0$.

An alternative representation is given by

$$F^{a,b}(Q, y) = a(y) \operatorname{Tr} Q + b(y) \sup_{|\mathbf{v}|=1, \mathbf{v}=0} \left\{\mathbf{v}^\mathsf{T} Q \mathbf{v}\right\}. \tag{2.8}$$

With the operator in this form, it is easy to see that when Q is negative definite, the operator is linear, and the operator homogenizes to the harmonic mean of a. The level sets of this

operator have corners on the negative axes and on the positive diagonal in the $\lambda_1 - \lambda_2$ plane. Elsewhere, the operator is linear in λ_1 and λ_2.

For the rest of the discussion we restrict to Q with at least one positive eigenvalue. In this case, we may restrict $\mathcal{A}$ to the set $\mathcal{A} = \{|\mathbf{v}| = 1\}$. Then the control set is determined by a single parameter: we parameterize $\mathbf{v}_\alpha = \begin{bmatrix}\sqrt{\alpha} & \sqrt{1-\alpha}\end{bmatrix}$, and write

$$B_\alpha := \mathbf{v}_\alpha \otimes \mathbf{v}_\alpha = \begin{bmatrix} \alpha & \sqrt{\alpha(1-\alpha)} \\ \sqrt{\alpha(1-\alpha)} & 1-\alpha \end{bmatrix}. \tag{2.9}$$

Then the linear component of (2.1) is $L_\alpha(Q, y) = [a(y)I + b(y)B_\alpha] : Q$ and $\mathcal{A} = [0, 1]$. This is the form of the operator which we use below.

Homogenization Formula 1 (Pucci with stripes)**.** Consider $F^{a,b}(Q, y)$ given by (3.19). Consider piecewise constant stripes, with

$$a(y_1) = 1, \qquad b(y_1) = \begin{cases} 0, & 0 \le y_1 \le \frac{1}{2} \\ b_0, & \frac{1}{2} \le y_1 \le 1 \end{cases} \tag{2.10}$$

For $Q \not\preceq 0$ and for $\alpha \in [0, 1]$ define

$$\overline{L_\alpha}(Q) = \operatorname{Tr} Q + \frac{b_0}{2 + b_0\alpha}\left(q_{22} + \alpha(q_{11} - q_{22}) + 2q_{12}\sqrt{\alpha - \alpha^2}\right),$$

Then

$$\overline{F^{a,b}}(Q) \ge \sup_{\alpha\in[0,1]} \overline{L_\alpha}(Q), \quad Q \not\preceq 0 \tag{2.11}$$

and

$$\overline{F^{a,b}}(Q) = \mathrm{HM}(a) \operatorname{Tr} Q, \quad Q \preceq 0.$$

Remark 2.3. This formula is obtained by considering constant controls in the direction of a given unit vector parameterized by α. We homogenize the linearization L_α about this control, and then optimize over the choice of α.

The term in (2.11) can be simplified further analytically, but the formula becomes complicated. It is more convenient to solve it numerically using one variable equation solvers.

Proof of Homogenization Formula 1. We need only consider the case $Q \not\preceq 0$ since the operator is linear otherwise.

1. Since we are on stripes, we restrict to invariant measures which depend only on y_1. The invariant measure must satisfy the constraint (2.5). For a choice of control $\alpha(y_1)$

depending only on y_1, this constraint is

$$\partial_{11}\left(c(y_1)p_{\alpha(y_1)}(y_1)\right) = 0,$$

where $c(y_1) := a(y_1) + b(y_1)\alpha(y_1)$. The solution is $p_{\alpha(\cdot)}(y_1) = \frac{\mathrm{HM}(c(y_1))}{c(y_1)}$.

2. Next with coefficients given by (2.10), restrict to $\alpha(y_1)$ to be constant on $b(y_1) = b_0$.

3. For these measures, the homogenized linear operator becomes

$$\overline{L_\alpha}(Q) = \mathrm{HM}(a)\,\mathrm{Tr}\,Q + \int_{\mathbb{T}^1} b(y_1)\,\mathrm{Tr}\,(B_\alpha Q)\,p_\alpha(y)dy \tag{2.12}$$

after integrating out the invariant measure from the Laplacian term.

Use the representation B_α given by (2.9) to simplify (2.12) to obtain the expression which is maximized in (2.11). □

2.2.2 *Maximum of two linear operators*

Definition 2.2.2. *Given a (constant) symmetric positive definite matrix, A, positive functions $a_0(y), a_1(y) > 0$, and the constant h. Define*

$$H(Q, y) = \max\{-a_0(y), -a_0(y) - a_1(y)\}A : Q + h \tag{2.13}$$

This operator may be written in HJB form (2.1) by writing

$$H(Q, y) = \max_{\alpha\in[0,1]} L_\alpha(Q, y), \quad L_\alpha(Q, y) \equiv -(a_0(y) + \alpha a_1(y))A : Q + h.$$

Homogenization Formula 2 (Maximum of two linear operators)**.** Let $H(Q, y)$ be given by (2.13). Then

$$\bar{H}(Q) = \max\{-\mathrm{HM}(a_0)A : Q, -\mathrm{HM}(a_0 + a_1)A : Q\} + h \tag{2.14}$$

Proof. 1. For any choice $\alpha(y)$, the corresponding invariant measure satisfying the constraint (2.5) is is given by

$$p_\alpha(y) = \frac{\mathrm{HM}(b(y, \alpha(y))}{b(y, \alpha(y))},$$

where $b(y, \alpha(y)) = a_0(y) + \alpha(y)a_1(y)$. The homogenized linear operator is then

$$\overline{L_{\alpha(y)}}(Q) = \int_{\mathbb{T}^d} L_{\alpha(y)}(Q, y)\,\mathrm{d}p_\alpha(y) = \mathrm{HM}(b(y, \alpha(y)))A : Q + h.$$

So from (2.3), we have

$$\overline{H}(Q) = \sup_{\alpha(\cdot)} \overline{L_{\alpha(y)}}(Q).$$

Notice that $b = (a_0(y) + \alpha a_1(y))$ is increasing in α for each y. Moreover, it is easy to verify that the Harmonic Mean is an increasing function of b. Thus, depending on the sign of $A : Q$, the optimal value is achieved by either $\alpha(y) \equiv 0$ or $\alpha(y) \equiv 1$, accordingly. This gives (2.14). □

2.2.3 *A one dimensional quadratic operator*

Definition 2.2.3. *In one dimension, with constants, c, $a > 0$ and the function $b(y) \geq 0$, consider*

$$H(Q, y) = aQ + b(y)(Q^+)^2 - c.$$

It is easy to verify that

$$H(Q, y) = \max_{\alpha \geq 0} L_\alpha(Q, y), \quad L_\alpha(Q, y) = A(y, \alpha)Q - h(y, \alpha)$$

with $A(y, \alpha) = a + 2b(y)\alpha$ and $h(y, \alpha) = b(y)\alpha^2 + c$.

Homogenization Formula 3. Let $H(Q, y)$ be given as in Definition 2.2.3. Suppose b is piecewise constant,

$$b(y) = \begin{cases} 0, & 0 \leq y \leq \frac{1}{2} \\ b_0, & \frac{1}{2} \leq y \leq 1 \end{cases} \tag{2.15}$$

Then

$$\overline{H}(Q) \geq a(Q + Q^+) - c + \frac{a^2}{b_0} - \frac{1}{b_0}\sqrt{a^3(a + 2b_0Q^+)}. \tag{2.16}$$

Proof. Consider constant controls $\alpha(y) \equiv \alpha$. In this case the invariant measure $p_\alpha(y)$ satisfying (2.5) is given by

$$p_\alpha(y) = \begin{cases} \dfrac{a + 2b_0\alpha}{a + b_0\alpha}, & 0 \leq y \leq \dfrac{1}{2} \\ \dfrac{a}{a + b_0\alpha}, & \dfrac{1}{2} \leq y \leq 1. \end{cases} \tag{2.17}$$

Then

$$\bar{L}_\alpha(Q) = \langle L_\alpha, p_\alpha \rangle \tag{2.18}$$

$$= \frac{a\,(a + 2b_0\alpha)}{a + b_0\alpha}\left[Q - \frac{1}{2}\left(\frac{c}{a} + \frac{b_0\alpha^2 + c}{a + 2b_0\alpha}\right)\right]. \tag{2.19}$$

By (2.3),

$$\overline{H}(Q) \geq \max_{\alpha \in [0,Q]} \overline{L_\alpha}(Q)$$

Next, maximize over α. This is accomplished by solving for the roots of the derivative of this expression with respect to α. We obtain

$$\alpha^*(Q) = \frac{1}{b_0}\left(-a + \sqrt{a(a + 2b_0Q^+)}\right). \tag{2.20}$$

Thus the estimate is given by (2.19) with α given by (2.20). Upon simplification, we obtain (2.16). □

2.2.4 *Separable operator in one dimension*

Definition 2.2.4. *Define a separable operator in one dimension as*

$$H(Q, y) = a(y)H_0(Q) \tag{2.21}$$

with $a(y) > 0$ *and* $H_0(Q) = \sup_\alpha \{\alpha Q - h(\alpha)\}$.

Homogenization Formula 4. Let $H(Q, y)$ be a separable operator in one dimension. Suppose also that $\alpha^* = \arg\max_\alpha \{\alpha Q - h(\alpha)\}$ is a singleton. Then

$$\overline{H}(Q) = \mathrm{HM}(a)H_0(Q).$$

Proof. From Theorem 3.2.2 we have that

$$\overline{H}(Q) = \sup_{\rho\in\Pr(\mathcal{A}\times\mathbb{T})} \left\{ \int_{\mathcal{A}\times\mathbb{T}} a(y)\left(\alpha Q - h(\alpha)\right) \mathrm{d}\rho \,\middle|\, (a(y)\rho)_{yy} = 0 \right\}. \tag{2.22}$$

In one dimension we can solve the equation (2.5) for the invariant measure ρ explicitly,

$$\rho(y, \alpha) = \frac{c(\alpha)}{a(y)} \tag{2.23}$$

where $c(\alpha)$ is chosen to ensure $\rho \in \Pr(\mathcal{A} \times \mathbb{T})$. That is, we need that $\int_{\mathcal{A}} c(\alpha)\,\mathrm{d}\alpha = \mathrm{HM}(a)$. With this constraint, (2.22) becomes

$$\overline{H}(Q) = \sup_{c(\alpha)} \left\{ \int_{\mathcal{A}} c(\alpha)\left(\alpha Q - h(\alpha)\right) \mathrm{d}\alpha \,\middle|\, \int_{\mathcal{A}} c(\alpha)\,\mathrm{d}\alpha = \mathrm{HM}(a) \right\} \tag{2.24}$$

$$\leq \mathrm{HM}(a) \sup_\alpha \left\{\alpha Q - h(\alpha)\right\}. \tag{2.25}$$

Equality is achieved by setting $c(\alpha) = HM(a)\delta(\alpha - \alpha^*)$.

□

2.3 NUMERICAL RESULTS

Here we compare the results of Section 2.2 with the numerical homogenization of the operators.

Remark 2.4 (Numerical methods). $\overline{H}(Q)$ was computed with two methods. In the first, the equation (2.1) was discretized with finite differences. A steady state solution was computed iteratively by Euler step to the parabolic equation $u_t + H(Q + D^2u, y)$. We used a filtered scheme [FO13] to choose between a monotone finite difference scheme and standard accurate finite differences. However the standard finite difference scheme was always chosen by the filtered scheme, likely because solutions are C^2 and periodic.

We also computed $\bar{H}_{LP}(Q)$ by discretizing the control space $\mathcal{A}$ and formulating the problem (2.6) as a discrete linear programming problem. Derivatives were discretized via standard second order finite differences. We then solved this LP using the package CVX [GB14, GB08] with the SeDuMi solver [Stu99].

Throughout we set $h(y, \alpha) = c = 1$, and subtracted this constant from $\overline{H}(Q)$, so as to avoid trivial solutions.

2.3.1 *Pucci type operator on stripes*

We compared the analytical formula, Homogenization Formula 1, with numerically homogenized values. This required solving a one-variable optimization problem. We used piecewise constant coefficients, where the operator was either $\operatorname{Tr} Q$, or $F^{1,2}(Q) = 1 \operatorname{Tr} Q + 2\lambda^+_{\max}(Q)$. The error profile of the analytic lower bound against the numerical homogenization is plotted in Figure 2.1a, for a set of diagonal Q. In the vicinity of the line $\lambda_1(Q) = \lambda_2(Q)$ the error is on the order of $1\mathrm{e}{-1}$; elsewhere the error is less than $1\mathrm{e}{-2}$.

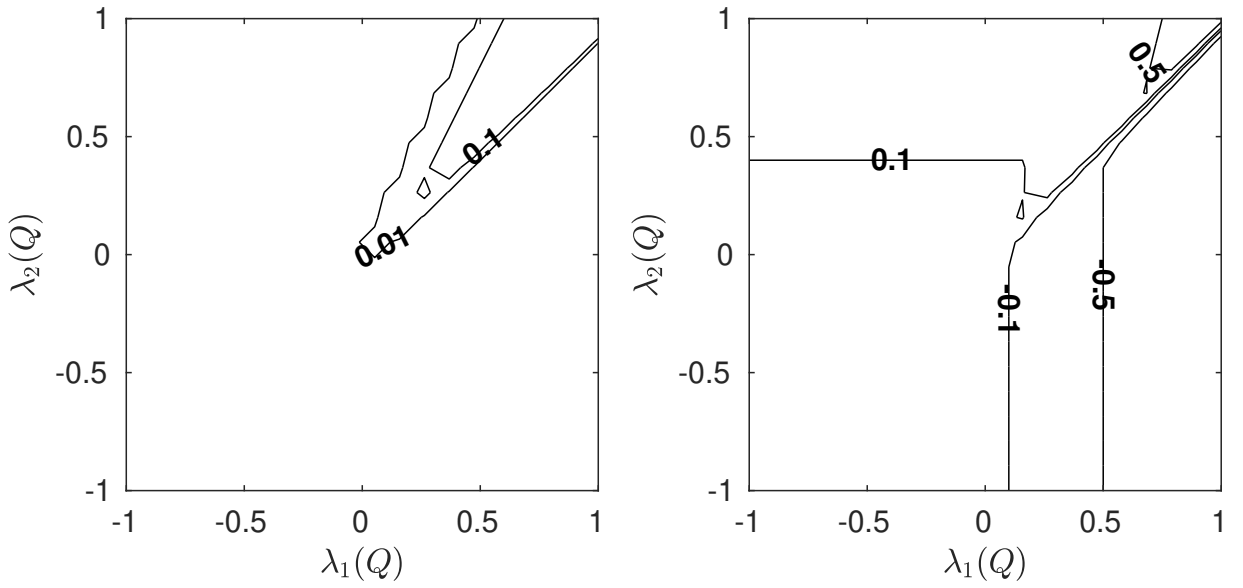
We contrast this homogenization approach with the method of homogenizing the linearized operator [FO18b]. The homogenizing error by first linearizing the operator is much greater than the error given by Formula 1, as can be seen by comparing Figures 2.1a and 2.1b.

There is a symmetry in $\bar{H}(Q)$. We represent

$$Q = R_\phi^T \operatorname{diag}(\lambda_1, \lambda_2) R_\phi$$

where R_ϕ is a rotation matrix. When $\phi = \pi/4$, the orientation of the stripes is at an equal angle to the eigenvectors of Q, then $\bar{H}(Q)$ is symmetric about $\lambda_1 = \lambda_2$. More generally $\bar{H}(Q)$ is symmetric under reflections in the angle about the same line of symmetry:

$$\bar{H}(Q|_{\phi=\pi/4-\gamma}) = \bar{H}(Q|_{\pi/4+\gamma}), \qquad \text{for } |\gamma| \le \pi/4.$$



a *Optimal Linearization error* **b** *Linearization error*

Figure 2.1: Comparison between homogenization error using an invariant measure, and by homogenizing the linearized operator. In this example the operator is given by (2.10), with $b_0 = 2$. Here $Q = \text{diag}(\lambda_1, \lambda_2)$. In the third quadrant, the operator is linear, and the error was zero up to machine precision. Figure 2.1a: error of Formula 1, the error is $1\text{e}{-8}$ is most of the domain, with the $1\text{e}{-2}$ level set shown. Figure 2.1b: error of homogenizing the linearized operator. There the error is order one, outside the third quadrant.

2.3.2 *Maximum of two linear operators, in one and two dimensions*

We numerically validated Homogenization Formula 2, for the maximum of two linear operators. We considered the case when the dimension is one, and we took $A = 1$. The interval $[0, 1]$ was discretized into 20 equal sized pieces. The coefficients $a_0(y)$ and $a_1(y)$ were piecewise constant on these equal-sized pieces. We took $h(y)$ to be constant. In Figure 2.2 we let a_0 alternate between 1 and $\frac{1}{2}$, and let a_1 alternate between $\frac{3}{2}$ and $\frac{5}{2}$. The values of the analytically homogenized operator are indistinguishable from the numerically computed values, for discrete values of Q, using both the direct method and the dual method. Even at the discontinuity $Q = 0$, the formula agrees with the numerical homogenization up to machine precision. For reference, we also plotted $H_1(Q) = \min_y H(Q, y)$ and $H_2(Q) = \max_y H(Q, y)$. We computed many different examples and obtained similar results. (Note in this example, the invariant measure is piecewise quadratic, so the numerical method is very accurate.) We also visualized the numerical invariant measure, and found that it agreed with our formula.

We also numerically validated Formula 2 in two dimensions, and obtained similar

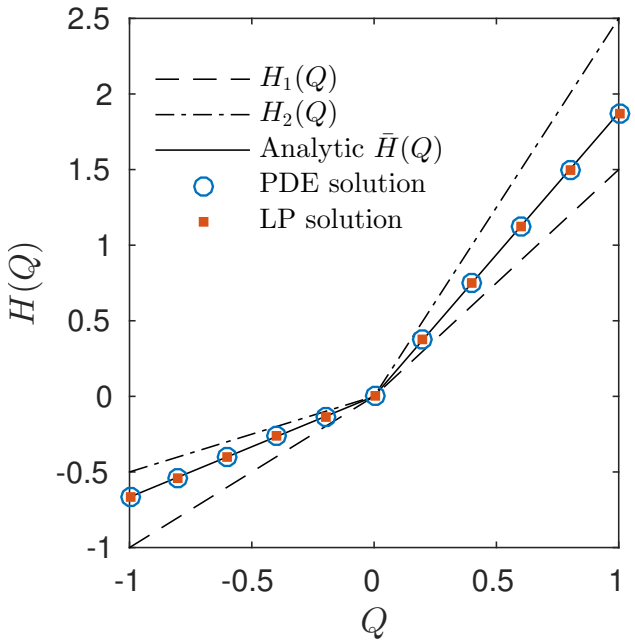


Figure 2.2: Validation of Formula 2, homogenization of the maximum of two linear operators, (2.13). Lines represent $\bar{H}(Q)$ and each of the operators $H_i(Q)$.

results: in this case the analytic formula and the numerical simulations agree up to $1e{-}12$.

2.3.3 *The quadratic operator*

Next we considered the example from §2.2.3, Homogenization Formula 3, for the operator

$$H(Q, y) = aQ + b(y)(Q^+)^2 - c.$$

Here c is a constant. We numerically homogenized this operator on the periodic domain $[0, 1]$, divided into 20 pieces. The coefficients are piecewise constant on equal intervals. As illustrated in Figure 2.3 the analytic homogenization and the numerically homogenized operator are indistinguishable. As in the previous operator (maximum of two linear operators), even at $Q = 0$ the formula agrees with the numerical homogenization up to machine precision. Again, we also plot $H_1(Q) = \min_y H(Q, y) = aQ$ and $H_2(Q) = \max_y H(Q, y) = aQ + b_0(Q^+)^2$.

2.4 NUMERICAL RATES OF CONVERGENCE IN THE PERIODIC AND RANDOM CASE

Using the exact analytical formulas of Section 2.2 (Formula 2 and Formula 3), we investigate empirical rates of convergence of the small-scale solutions u^ε to the solution $\bar{u}$ of the homogenized operator. Although our theoretical results were for the periodic case, we

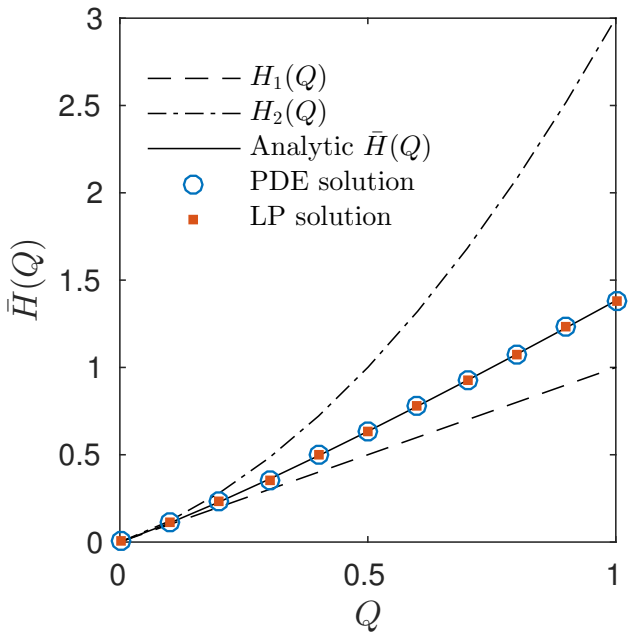


Figure 2.3: Validation of Formula 3. Value of $H_1(Q), H_2(Q)$, the numerically homogenized operators, and the analytic homogenized operator.

found that the same formulas applied in the random case. This allows us to study empirical convergence rates in the random case as well.

We solved the Dirichlet problem with zero boundary conditions on the interval $[0, 1]$ for the two different operators, in both the random and periodic case. The operators were the maximum of two linear operators, Formula 2; and the quadratic operator, Formula 3. These are both one dimensional examples.

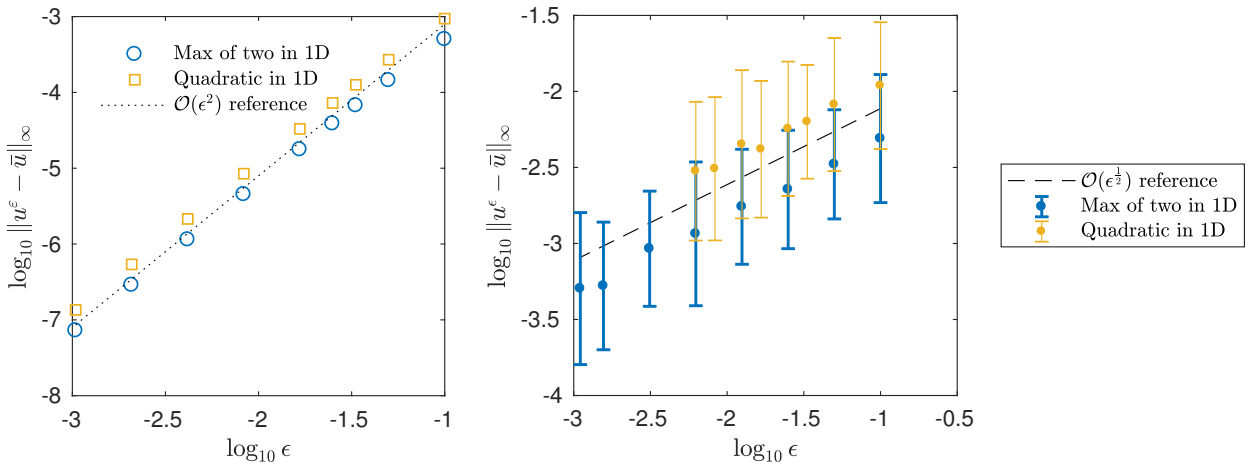
We used a sequence of decreasing cell sizes, of width ε. We used 1 grid point per cell.

We also solved the same problem with the homogenized operator. Numerically we obtained two solutions, u^ε, and $\bar{u}$ corresponding to

$$\begin{cases} H^\varepsilon(D^2u^\varepsilon(x), x) = 1 \\ u^\varepsilon(x) = 0, \quad x \in \partial[0,1] \end{cases} \quad \text{and} \quad \begin{cases} \bar{H}(D^2\bar{u}(x), x) = 1 \\ \bar{u}(x) = 0, \quad x \in \partial[0,1]. \end{cases} \tag{2.26}$$

We chose coefficients which were piecewise constant. Let $H^\varepsilon(Q, x)$ be the operator parameterized by checkerboard square width ε. We checked convergence for both the periodic case, and the random case. In the periodic case, the unhomogenized operator alternates between two constituent operators H_1 and H_2 between the checkerboard cells. In the random case, in each checkerboard square we randomly sample from the two constituent operators with probability $\frac{1}{2}$.

In the random case, we observed convergence rates consistent with $\mathcal{O}(\varepsilon^{\frac{1}{2}})$ in the sup-



a *Periodic convergence* **b** *Random convergence*

Figure 2.4: Figure 2.4a: Periodic coefficients: rate of convergence $u^\varepsilon \to \bar{u}$. Figure 2.4b: Random coefficients: rate of convergence. We plot 90% confidence intervals for a normal distribution.

Table 2.1: Empirical rates of convergence for the two operators.

Operator	Periodic, $\|\cdot\|_\infty$	Random, $\|\cdot\|_\infty$	Random, $\|\cdot\|_2$	Random, $\|\cdot\|_1$
Max of two in 1D	1.95	0.51	0.49	0.50
Quadratic in 1D	1.95	0.48	0.42	0.41

norm for both operators. In the periodic case, we observed convergence rates consistent with $\mathcal{O}(\varepsilon^2)$.

Figure 2.4 presents the observed rates of convergence as $u^\varepsilon \to \bar{u}$ in the sup-norm. In the periodic setting, the order is nearly $\mathcal{O}(\varepsilon^2)$: we estimate that the order of convergence is $\mathcal{O}(\varepsilon^{1.95})$. In the random setting, we solved each problem 20 times, drawing the random checkerboard anew at each iteration. We then used least squares to estimate the order of convergence. We summarize these convergence estimates in Table 2.1. It appears that convergence in the sup-norm is roughly $\mathcal{O}(\varepsilon^{\frac{1}{2}})$ on the random checkerboard. We also measured the errors in the ℓ^2 and ℓ^1 norms.

2.5 CONCLUSIONS

In this article we investigated the accuracy of approximating nonlinear homogenization by the homogenization of a linearization of the operator. In previous work [FO18b], we simply linearized about a constant. There, we obtained very accurate for checkerboard

type coefficients, but significant errors in the case of stripes. In this article, we restricted to convex operators. This allowed us to write operators as the supremum of linear operators. For any linearization over a choice of control $\alpha(y)$

$$\overline{L_\alpha}(Q) \leq \overline{H}(Q)$$

with equality when $\alpha(y)$ is optimal.

We applied this formula to three examples. For the example of a maximum of two linear operators, we obtained an exact result, given by the maximum of two harmonic means (see (2.14)). For a quadratically nonlinear one dimensional operator, we restricted to piecewise constant controls and optimized over the value of the control. This results in a lower bound which was verified by numerical simulations to be exact.

Finally, we consider the Pucci-type operator with stripe coefficients. In this case, the controls depended on a choice of direction vector, which in two dimensions resulted in a one parameter optimization problem for $\overline{L_\alpha}(Q)$. The solution of this problem was verified by numerical simulations to be nearly exact over parameter values away from the singularities of the operator. For other values of Q it achieved a small (a few percentages) relative error.

We also consider the numerical convergence rates of the homogenization problem in the scale parameter, obtaining results consistent with recent analytical results, in both the periodic and random case.

CHAPTER 3

APPROXIMATE HOMOGENIZATION OF FULLY NONLINEAR ELLIPTIC PDES: ESTIMATES AND NUMERICAL RESULTS FOR PUCCI TYPE EQUATIONS

Abstract

We are interested in the shape of the homogenized operator $\overline{F}(Q)$ for PDEs which have the structure of a nonlinear Pucci operator. A typical operator is $H^{a_1,a_2}(Q,x) = a_1(x)\lambda_{\min}(Q) + a_2(x)\lambda_{\max}(Q)$. Linearization of the operator leads to a non-divergence form homogenization problem, which can be solved by averaging against the invariant measure. We estimate the error obtained by linearization based on semi-concavity estimates on the nonlinear operator. These estimates show that away from high curvature regions, the linearization can be accurate. Numerical results show that for many values of Q, the linearization is highly accurate, and that even near corners, the error can be small (a few percent) even for relatively wide ranges of the coefficients.

3.1 INTRODUCTION

In this article we consider fully nonlinear, uniformly elliptic PDEs $F(Q,x)$. We are interested in approximating the homogenized operator $\overline{F}(Q)$. We focus on Pucci-type PDE operators in two dimensions. The restriction to two dimensions is for computational simplicity and also for visualization purposes. We consider periodic coefficients, although in our numerical experiments we obtained very similar results with random coefficients.

The approach we take is to linearize the operator about the value Q, and to homogenize the linearized operator $\overline{L}(Q)$. The solution of the linear homogenization problem can be expressed (and in some cases solved analytically) by averaging against the invariant measure. The result is given by

$$\overline{L^Q}(Q) = \int F(Q,x)\rho^Q(x)dx$$

where ρ^Q is the invariant measure of the corresponding linear problem. We estimate the linearization error

$$E(Q) \equiv \overline{F}(Q) - \overline{L^Q}(Q),$$

For convex operators, the analysis gives a one sided bound on the error. In general, we obtain upper or lower bounds on the error, which depend on generalized semi-concavity/convexity estimates of F, as well as on the solution of the cell problem u^Q for the nonlinear problem. These results are stated in Theorem 3.2.2 below.

For theoretical results on nonlinear homogenization, we refer to the review [ES08] as well as recent works on rates of convergence (for example [AS14]). There are fewer works which aim to determine the values $\overline{F}(Q)$. Few analytical results are available. Numerical homogenizing results for Pucci type operators can be found in [CG08] using a least-squares formulation. We also mention numerical work by [GO04] and [OTV09] and [LYZ11] in the first order case, as well as [FO09] in the second order linear non-divergence case.

The typical operator we consider herein is defined next. Below, we consider more operators, including the usual convex Pucci Maximal operator.

Definition 3.1.1 (Fully nonlinear elliptic operator $F(Q,x)$ and linearization)**.** *We are given $F : \mathcal{S}^d \times \mathbb{T}^d \to \mathbb{R}$ which is uniformly elliptic, Lipschitz continuous in the first variable and bounded in the second variable. Suppose for a given Q, that $\nabla_Q F(Q,x)$ exists for all x. Write*

$$L^Q(M,x) = \nabla_Q F(Q,x) \cdot (M-Q) + F(Q,x) \tag{3.1}$$

for the affine approximation to F at Q.

Given $Q \in \mathcal{S}^d$, write, for $d = 2$, $\lambda_{\min}(Q)$ and $\lambda_{\max}(Q)$ for the smaller, and larger eigenvalues of Q, respectively.

Example 3.1 (Typical PDE operator)**.** Given $\delta > 0$, and periodic functions $a_1(y), a_2(y) \geq \delta$. Define the homogeneous order one PDE operator

$$H^{a_1,a_2}(Q,x) = a_1(x)\lambda_{\min}(Q) + a_2(x)\lambda_{\max}(Q) \tag{3.2}$$

Suppose Q has unit eigenvectors v_1, v_2 corresponding to the eigenvalues $\lambda_{\min}(Q), \lambda_{\max}(Q)$, respectively. Then the linearization at Q, of H^{a_1,a_2} is given by

$$L^Q(M,x) = a_1(x)v_1^T M v_1 + a_2(x)v_2^T M v_2 \tag{3.3}$$

Remark 3.1 (Typical results). We consider the case of coefficients which are either (i) periodic checkerboards or (ii) random checkerboards. We compute both the nonlinear homogenization $\overline{F}(Q)$ and the homogenized linear operator $\overline{L^Q}(Q)$. In practice, the numerically computed error is insignificant, less than $1\mathrm{e}{-8}$ for values of Q, away from regions of high curvature of F with respect to Q. Areas where the error is significant correspond to regions where the semi-concavity constants are large. A typical result is displayed in

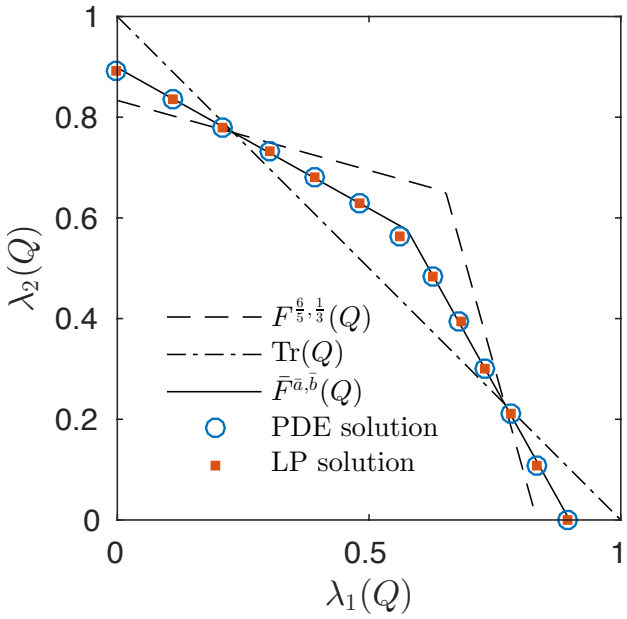


Figure 3.1: Plot of a single level set of $\overline{L^Q}(Q)$ and $\overline{F}(Q)$. This example is typical. In this case the coefficients are on a random checkerboard. The error is only visible near the corner of the level set of the operator. $F(Q, y)$ a Pucci-type operator, see Definition 3.3.1 below. The details of the coefficients can be found in Section 3.4.

Figure 3.1. The solid line is a level set of the homogenized linear operator $\overline{L^Q}(Q)$. The dots are numerical computations of $\overline{F}(Q)$. The error is very small, except at one point, which corresponds to a corner of the operator. (Dashed lines indicate underlying operators which comprise $F(Q, x)$.

Our analysis depends on the shape of $F(Q, x)$ in Q, but not on the pattern of the coefficients in x. We also considered the case of stripe coefficients. For separable examples, the linear approximation is still effective. However, we also found nonseparable examples where the linear approximation is poor, which we will address in a companion paper [FO18a] with a closer bound.

3.1.1 *Background: cell problem and linear homogenization*

In this section, we review background material on the cell problem for the nonlinear PDE, and on linear homogenization. We also give an exact formula in one dimension for a separable operator.

Given $a(y) : \mathbb{T} \to \mathbb{R}$, positive, $a(y) > 0$, write $\mathrm{HM}(a) = \left(\int_{\mathbb{T}} \frac{\mathrm{d}y}{a(y)}\right)^{-1}$ for the harmonic mean of a.

In the linear case, $\overline{L}$ can be found by averaging against the invariant measure, by solving the adjoint equation (see [BLP11] or [FO09]), which yields the following formula.

Lemma 3.1.2 (Linear Homogenization Formula)**.** *The separable linear operator* $L(M,x) = a(x)A_0 : M + f(x)$ *has invariant measure* $\rho(x) = \mathrm{HM}(a)/a(x)$ *and homogenizes to* $\overline{L}(Q) = \mathrm{HM}(a)A_0 : Q + \overline{f}$*, where* $\overline{f}(x) = \int f(x)\,\mathrm{d}\rho(x)$*.*

For the nonlinear operator F, the homogenized operator is given by solving the cell problem, see [Eva89].

Definition 3.1.3 (Solution of the cell problem)**.** *Given* F *uniformly elliptic, for each* $Q \in \mathcal{S}^d$*, there is a unique value* $\overline{F}(Q)$ *and a periodic function* $u^Q(y)$ *which is a viscosity solution of the cell problem*

$$F(Q + D^2u^Q(y), y) = \overline{F}(Q). \tag{3.4}$$

Lemma 3.1.4 (Homogenization of linearized operator)**.** *Consider the nonlinear elliptic operator* $F(Q,x)$*, and suppose for a given* Q*, that* $\nabla_Q F(Q,x)$ *exists for all* x*. The corresponding linearization at* Q *is given by* (3.1)*. Let* ρ^Q *be the corresponding unique invariant probability measure, which is the solution of the adjoint equation*

$$D^2 : (\nabla_Q F(Q,y)\rho^Q(y)) = 0, \tag{3.5}$$

interpreted in the weak sense. Then $\overline{L^Q}(Q)$*, the homogenized linearized operator evaluated at* Q*, is given by*

$$\overline{L^Q}(Q) = \int_{\mathbb{T}^d} F(Q,y)\,\mathrm{d}\rho^Q(y). \tag{3.6}$$

Proof. The invariant measure ρ^Q solves (3.5), see [BLP11] or [FO09]. Apply (3.1) at $M = Q$ and then integrate against ρ^Q to obtain the result. □

3.2 MAIN RESULT

3.2.1 *Generalized semiconcavity estimates on the operators*

Consider the uniformly elliptic operator $F(Q,x)$, where $Q \in \mathcal{S}^d$ and $x \in \mathbb{T}^d$. We assume the following.

Assumption 3.2.1 (Quadratically dominated for $F(Q,x)$)**.** Let F be as in Definition 3.1.1. Suppose for a given Q, that $\nabla_Q F(Q,x)$ exists for all x. Write $\|Q\|$ for the Frobenious norm of Q. We say that F is *quadratically dominated above* at Q if there is a bounded function $C^+(Q,x) : \mathbb{T}^d \to \mathbb{R}$ such that

$$F(M,x) - L^Q(M,x) \le C^+(Q,x)\frac{\|M-Q\|^2}{2}, \quad \text{for all } (M,x) \in \mathcal{S}^d \times \mathbb{T}^d \tag{3.7}$$

and similarly, F is *quadratically dominated below* at Q if there is a bounded function $C^-(Q,x):\mathbb{T}^d\to\mathbb{R}$ such that

$$F(M,x)-L^Q(M,x)\geq C^-(Q,x)\frac{\|M-Q\|^2}{2},\qquad \text{for all } (M,x)\in\mathcal{S}^d\times\mathbb{T}^d \tag{3.8}$$

Remark 3.2. If F is convex in Q, then $C^-(Q,y)=0$. Similarly if F is concave in Q, $C^+(Q,y)=0$. More generally if F is semi-concave, or semi-convex in Q, then we can set $C^\pm(Q,y)=C^\pm(y)$, to be a constant independent of Q. However, we require the definition above for when the semi-concavity or semi-convexity conditions in Q do not hold, as is the case for the Pucci-type operators defined below.

Example 3.2. Let $x\in\mathbb{R}$ and set $f(x)=\max\{ax,bx\}$. Since f is convex, we can take $C^-(x)=0$ in (3.8). We claim that for $x\neq 0$, (3.7) holds with

$$C^+(x)=\frac{|a-b|}{2x}, \tag{3.9}$$

and this is the best constant. See Figure 3.2.

Derivation of (3.9). Expand $f(x+y)$ about the point x, for $x\neq 0$. To test (3.7), replace the inequality with an equality to obtain a quadratic equation. By requiring that there is only one root, we obtain an equation for the discriminant of the quadratic, which can be solved to obtain the result. □

3.2.2 *Main Theorem*

Theorem 3.2.2. *Suppose F satisfies Assumptions 3.2.1 and $u^Q\in C^{2,\alpha}(\mathbb{T}^d)$ is a classical solution. Let $\overline{F}(Q)$ be the homogenized operator at Q and let u^Q be the corresponding solution of the cell problem given by (3.4). Let the homogenization of the linearization of the operator be given by (3.6) and let $\rho^Q(y)$ be the corresponding invariant measure of the linearized problem (3.1). Write*

$$\overline{C^\pm}(Q)=\frac{1}{2}\int C^\pm(Q,y)\|D^2u^Q(y)\|^2\,\mathrm{d}\rho^Q(y)$$

Then

$$\overline{C^-}(Q)\leq\overline{F}(Q)-\overline{L^Q}(Q)\leq\overline{C^+}(Q) \tag{3.10}$$

Remark 3.3. In the examples we consider below, $C^\pm(Q,y)\to 0$ as $\operatorname{dist}(|Q|,S)\to\infty$, for the singular set of the operator. This gives control over the homogenization error for many values of Q. Another term in the error is $\|D^2u^Q\|$. In the homogeneous order one case, we have $u^Q=0$ for $Q=0$, so a continuity argument suggests that we may have control

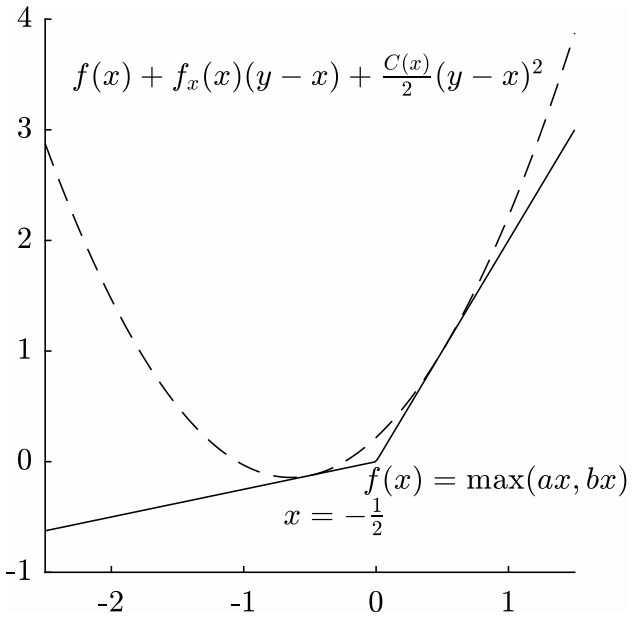


Figure 3.2: For the simple example $f(x) = \max\{ax, bx\}$, the semi-concavity constant is $C(x) = C^+(x) = \frac{|a-b|}{2x}$.

of $\|D^2 u^Q\|$ for small values of Q. This is the case in one dimension in [FO18a], where we obtain an analytical formula for u^Q_{xx} through (3.16), which gives $|u^Q_{xx}| \leq C|Q|$.

The main theorem is a formal result in the sense that it relies on the fact that u^Q is a classical solution, which does not hold in general. If F is convex (or concave), then by a famous theorem of Krylov and Evans [Kry84, Eva82], or [CC95], $u^Q \in C^2(\mathbb{T}^d)$. However, in general we are only guaranteed $u^Q \in C^{1,\alpha}(\mathbb{T}^d)$ [Jen88].

Proof. Subtract the linearization of F at Q evaluated at $Q + D^2 u^Q(y)$ from the equation for the cell problem (3.4), to obtain

$$\overline{F}(Q) - L^Q(Q + D^2 u^Q, y) = F(Q + D^2 u^Q, y) - L^Q(Q + D^2 u^Q, y). \tag{3.11}$$

From Assumption (3.2.1),

$$\overline{F}(Q) - L^Q(Q + D^2 u^Q, y) \leq C^+(Q, y)\frac{\|D^2 u^Q\|^2}{2} \tag{3.12}$$

and

$$\overline{F}(Q) - L^Q(Q + D^2 u^Q, y) \geq C^-(Q, y)\frac{\|D^2 u^Q\|^2}{2}. \tag{3.13}$$

Now integrate eqs. (3.12) and (3.13) against the invariant measure ρ^Q. This yields the upper

and lower bounds (3.10), where we have used the fact that for all $\phi \in C^2(\mathbb{T}^d)$,

$$\int_{\mathbb{T}^d} L^Q(Q + D^2\phi, y)\, \mathrm{d}\rho^Q(y) = \int_{\mathbb{T}^d} F(Q, y)\, \mathrm{d}\rho^Q(y), \tag{3.14}$$

which follows from integration by parts, since ρ^Q solves the adjoint equation (3.5). □

3.2.3 *Applications of the main result*

We give two applications of the main result. In the first example, where the operator is separable, we have an analytical formula for $\overline{L^Q}(Q)$. In this case the estimates also simplify. In the second, nonseparable example, we can find $\overline{L^Q}(Q)$ by solving a single linear homogenization problem, with coefficients given by the linearization (3.3).

Corollary 3.2.3. *Consider the separable, purely second order operator*

$$F(Q, y) = a(y)F_0(Q)$$

for $y \in \mathbb{R}^d$. Suppose that F_0 is quadratically dominated with constants $C^-(Q)$ and $C^+(Q)$. Then,

$$\overline{C^-}(Q) \leq \overline{F}(Q) - \mathrm{HM}(a)F_0(Q) \leq \overline{C^+}(Q) \tag{3.15}$$

where

$$\overline{C^\pm}(Q) = \frac{1}{2}C^\pm(Q)\,\mathrm{HM}(a)\int_{\mathbb{T}^d} \|D^2u^Q(y)\|^2 \frac{1}{a(y)}\,\mathrm{d}y.$$

Proof. 1. The formula for the linearization,

$$\overline{L^Q}(Q) = \mathrm{HM}(a)F_0(Q)$$

follows from the Linear Homogenization Formula (Lemma 3.1.2).

2. From linearization, we have that $\rho^Q(y) = \mathrm{HM}(a)/a(y)$. Using the definition, then the generalized semiconvexity/concavity constants for $F(Q, y)$ are given by

$$C^+(Q, y) = a(y)C^+(Q), \quad \text{and} \quad C^-(Q, y) = a(y)C^-(Q).$$

Passing the constants and the invariant measure into Theorem 3.2.2 gives the bounds provided by (3.15), since the coefficients $a(y)$ cancel.

□

Remark 3.4. In a companion paper [FO18a], we show that for convex operators in one dimension,

$$\overline{F}(Q) = \overline{L^Q}(Q) = \mathrm{HM}(a)F_0(Q). \tag{3.16}$$

Corollary 3.2.4. *Consider the operator H^{a_1,a_2} given by (3.2). Then*

$$\left|\overline{H^{a_1,a_2}}(Q) - \overline{L^Q}(Q)\right| \leq \frac{1}{2|\lambda_{\min}(Q) - \lambda_{\max}(Q)|} \int |a_1(y) - a_2(y)| \, \|D^2 u^Q(y)\|^2 \, d\rho^Q(y)$$

Proof. We apply Theorem 3.2.2 to H^{a_1,a_2} given by (3.2). The linearization is given by (3.3). The invariant measure of the linear problem is given by the solution of (3.5) and the homogenized linear operator is given by (3.6) from Lemma 3.1.4.

The main step is to work out the generalized semi-concavity constants. We claim.

$$C^+(Q,x) = \frac{(a_2(x) - a_1(x))^+}{|\lambda_{\min} - \lambda_{\max}|}, \quad \text{and} \quad C^-(Q,x) = \frac{(a_1(x) - a_2(x))^-}{|\lambda_{\min} - \lambda_{\max}|}. \tag{3.17}$$

To prove this we proceed in steps.

1. First, take $q \in \mathbb{R}^2$ and set $f(q) = \max(q_1, q_2)$. Then $L^q(y) = \nabla f(q) \cdot y$ away from the singular set $q_1 = q_2$, since the function is homogeneous of order one. The constant $C^-(q) = 0$, since f is convex. We claim the optimal choice for $C^+(q)$ is given by

$$C^+(q) = \frac{1}{|q_2 - q_1|}$$

for $q_1 \neq q_2$. To see this, we require

$$\max(y_1, y_2) \leq \nabla f(q) \cdot y + \frac{C^+(q)}{2} |y - q|^2.$$

It is easily verified that the extremal case for the inequality occurs when $(y_1, y_2) = (q_2, q_1)$, which leads to the condition

$$|q_1 - q_2| \leq C^+(q) |q_1 - q_2|^2.$$

giving the result.

2. Let $f(q_1, q_2) = a_1 \min(q_1, q_2) + a_2 \max(q_1, q_2)$. Rewrite $f(q_1, q_2) = a_1(q_1 + q_2) + (a_2 - a_1) \max(q_1, q_2)$. We can always subtract an affine function when computing the constants. So the constants for f are the same as the constants for $(a_2 - a_1) \max(q_1, q_2)$. In this case, using the result of step 1, we obtain

$$C^+(x) = \frac{(a_2 - a_1)^+}{|q_1 - q_2|}, \qquad C^-(x) = \frac{(a_1 - a_2)^-}{|q_1 - q_2|}$$

3. Next consider for the two by two matrix Q, $h(Q, x) = a_1(x) \min(q_{11}, q_{22}) + a_2(x) \max(q_{11}, q_{22})$. Then the previous step shows that the constants for h are given by the previous ones (with

q_{11} replacing q_1 and q_{22} replacing q_2. Finally, since H^{a_1,a_2} depends only on the eigenvalues of Q, without loss of generality, we can choose a coordinate system where Q is diagonal when computing the generalized semiconcavity constants. It remains to show that the generalized semi-concavity condition holds for a matrix, M. If M is diagonal the condition holds. But if M is not diagonal, then the change in the norm $\|M - Q\|^2$ can be controlled by a constant, or absorbed into the definition of the norm. □

3.3 Computational Setting

For our numerical experiments, we consider a wider class of separable and non-separable operators.

3.3.1 *PDE Operators*

Definition 3.3.1 (Pucci-type operators)**.** *For $\delta > 0$ and given functions $0 < \delta \leq a(y) \leq A(y)$. Write $b(y) = A(y) - a(y)$. Also write $t^+ = \max(t, 0)$. Define, for $d = 2$, the standard Pucci maximal operator, the Pucci-type operator, the smoothed Pucci-type operator, and a Monge-Ampere type operator respectively as*

$$P^{A,a}(Q, y) = a(y) \operatorname{Tr} Q + b(y) \left(\lambda^+_{\min}(Q) + \lambda^+_{\max}(Q)\right) \tag{3.18}$$

$$F^{A,a}(Q, y) = a(y) \operatorname{Tr} Q + b(y)\lambda^+_{\max}(Q). \tag{3.19}$$

$$F^{A,a}_k(Q, y) = a(y) \operatorname{Tr} Q + b(y)\mathcal{S}_k(\lambda_{\min}(Q), \lambda_{\max}(Q), 0) \tag{3.20}$$

$$M(Q, y) = a(y) \left(\operatorname{Tr}(Q) + \lambda^+_{\min}(Q)\lambda^+_{\max}(Q)\right). \tag{3.21}$$

Here, for $x \in \mathbb{R}^m$, $S_k(x)$ is the smoothed maximum function,

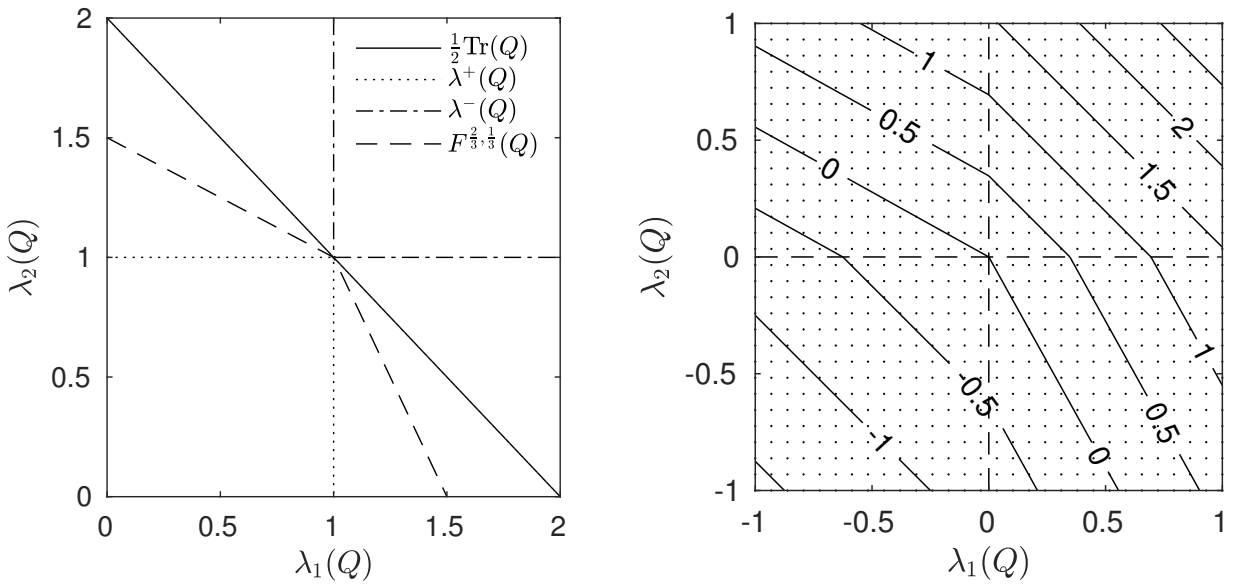
$$\mathcal{S}_k(x) = \frac{\sum_{i=1}^m x_i \exp(kx_i)}{\sum_{i=1}^m \exp(kx_i)}. \tag{3.22}$$

The function S_k goes to the max as $k \to \infty$, and to the average as $k \to 0$.

Definition 3.3.2 (Periodic checkerboard, stripes, and random checkerboard coefficients)**.** *Define*

$$a_0(y) = \begin{cases} 1, \ y \in B \\ r, \ y \in W, \end{cases} \tag{3.23}$$

with $r > 1$. The sets B and W are either black and white squares of a checkerboard; alternating black and white stripes of equal width; or a 'random' checkerboard, with black and white squares distributed with equal probability, uniformly.



a *Example level sets* **b** *Pucci level sets*

Figure 3.3: Figure 3.3a: level set plot of several operators as function of the eigenvalues of Q. Figure 3.3b: Level sets of an example Pucci operator, $P^{\frac{5}{4},\frac{2}{3}}(Q)$. Points indicate values of Q that were homogenized.

Remark 3.5 (Representation and visualization of the operators). The definition above agrees with the usual definition of the Pucci operator,

$$P^{A,a}(Q,y) = \sup\{M : Q \mid a(y)I \ll M \ll A(y)I\}. \tag{3.24}$$

We can also rewrite

$$F^{A,a}(Q,y) = \begin{cases} a(y)\,\mathrm{Tr}\,Q & Q \text{ negative definite} \\ H^{A,a}(Q,x), & \text{otherwise.} \end{cases}$$

Example 3.3. For the Pucci operator $P^{A,a}(Q,x)$ given by (3.18), by convexity, $C^-(Q,x) = 0$ and

$$C^+(Q,x) = \begin{cases} \max\left\{\frac{b(x)}{\mathrm{Tr}(Q)}, \frac{A(x)}{2\lambda_{\min}}\right\}, & \text{if } \lambda_{\min}, \lambda_{\max} > 0 \\ \frac{b(x)}{2\min(|\lambda_{\min}|,|\lambda_{\max}|)}, & \text{otherwise.} \end{cases} \tag{3.25}$$

3.3.2 *Numerical Method details*

In order to compute the errors, as a function of Q, we used a grid in the $\lambda_1 - \lambda_2$ plane, and computed the linear and nonlinear homogenization at the grid points. Typical values can be see in Figure 3.3b, where black points indicate the grid values of Q tested.

We remark on the numerical methods used throughout. To compute $\overline{F}(Q)$ directly, we discretized with finite differences and solved the parabolic equation $u_t + F(Q + D^2u, y)$ using an explicit Euler method, to iteratively compute a steady state solution. We discretized using a convergent monotone scheme [Obe06] and also using standard finite differences. The accuracy of the monotone scheme was less than the standard finite differences, so we implemented a filtered scheme [FO13]. In practice, the filtered scheme always selected the accurate scheme, so in this instance, perhaps because the solutions are C^2 and periodic, standard finite differences appear to converge.

For all the computations, to avoid trivial solutions, we solved with a right hand side function equal to a constant, and then subtracted the same constant from $\overline{F}(Q)$.

The computational domain was the torus $[0, 1]^2$, divided into 20×20 equal squares, each with 16 grid points per square.

Remark 3.6 (Comparison with [CG08]). The problem of homogenizing $a_0(y)F^{A,a}(Q)$, was considered in [CG08]. In their case, the spatial coefficient $a_0(y)$ varies periodically and smoothly between 2 and 3, and their homogenized value for $\overline{a}_0$ was 2.5 (which was the average of the coefficient $a_0(x, y) = \cos(\pi x)\cos(\pi y)$). Our results using these coefficients was $\overline{a}_0 = 2.486$, which is very close to the average. However with coefficients which are more spread out, we obtain values far from the average.

3.4 Numerical results

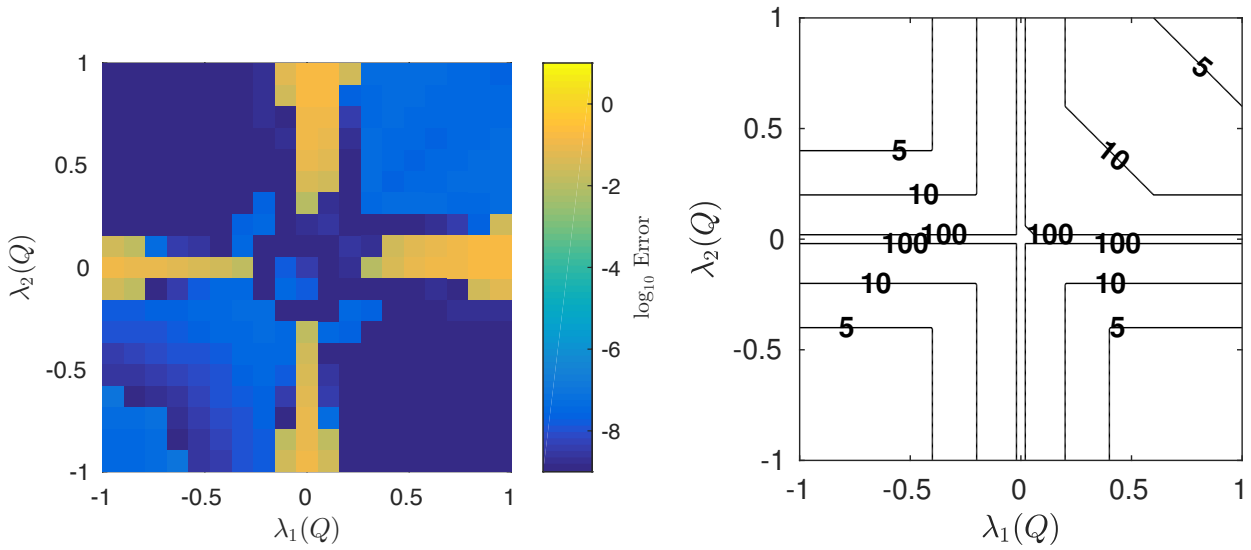
3.4.1 *Numerical Results: separable operators*

Here we check the homogenization error of the bound for separable operators in two dimensions, from Corollory 3.2.3. We are in the convex case, so the lower bound is zero.

We performed numerical simulations on four operators, see Definition 3.3.1.

- $a_0(y)P^{3,1}(Q)$
- $a_0(y)F^{3,1}(Q)$
- $a_0(y)F_k^{3,1}(Q)$, with $k = 10$ and $k = 0.1$
- $a_0(y)M(Q)$

In Figure 3.4 we compare the error $\overline{F}(Q) - \overline{L}^Q(Q)$ for a separable Pucci operator on a checkerboard, we also illustrate the constant $C^+(Q, y)$. In this case we have an analytical formula for $\overline{L}^Q(Q)$). This figure illustrates the Main Theorem: when the constant is large

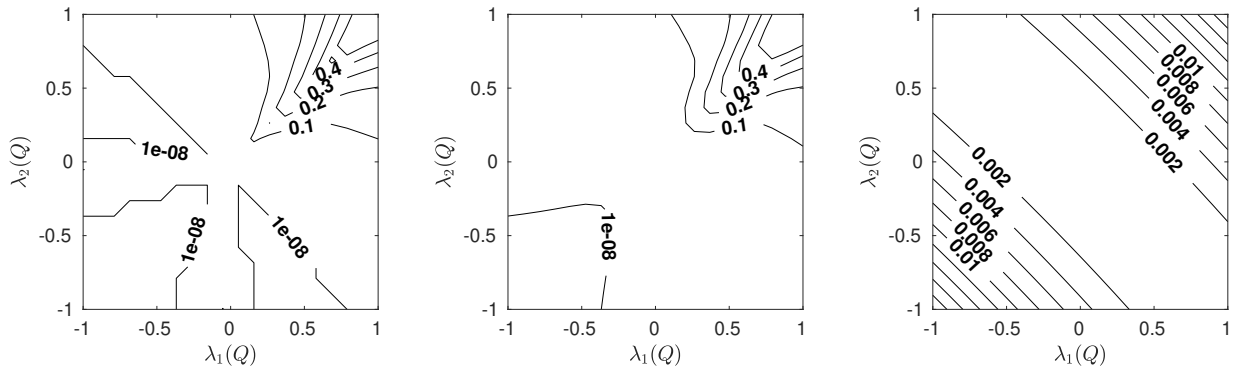


a *Error* **b** $\max_y C^+(Q, y)$

Figure 3.4: Homogenization of a separable Pucci example operator, $a(y)P^{3,1}$, on a periodic checker board, with coefficients of 1 or 2 ($r = 2$). 3.4a: Error $\overline{F}(Q) - \overline{L}^Q(Q)$. Figure 3.4b: An upper bound of the semi-concavity constant $C^+(Q, y)$. The error is $1\mathrm{e}{-6}$ or less in the blue part of the domain. In the yellow region it goes from 0.01 up to 0.15. The regions where the error is small coincide with smaller values of the semi-concavity constant.

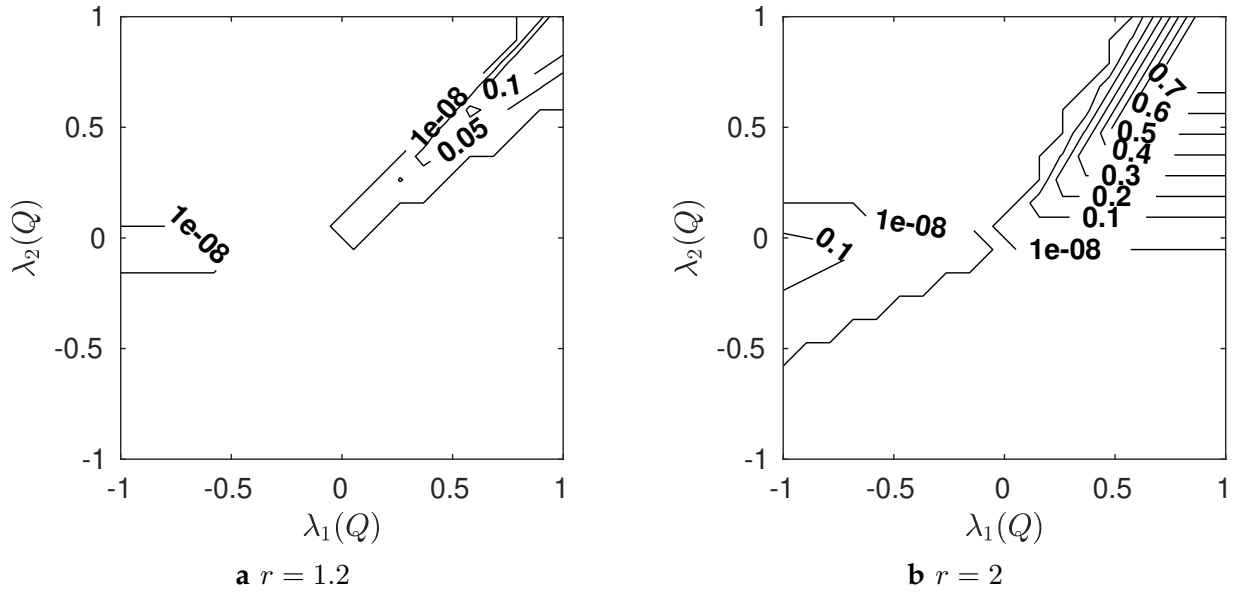
the error from the linearization is high. The error is less than $1\mathrm{e}{-6}$ outside of a small region about the axis, and on the order of 0.1 near the axis.

In Figure 3.5, we show how the error $\overline{F}(Q) - \overline{L}^Q(Q)$ decreases as the operator becomes smoother. The operator with the smallest maximum curvature (Figure 3.5c) exhibits the smallest error. As the operator becomes less smooth, the error increases. For the smoothest operator the global error is at most one percent (in the range of values shown in the figure). For the two sharper operators, there is still very high accuracy away from the highest curvature regions. We see that error of the smooth operator, $F_{10}^{3,1}(Q)$, is slightly smaller than the non smooth operator's error near the line $\lambda_1 = \lambda_2$ (this is where the non smooth operator is not differentiable). As the smoothing constant $k \to 0$, the error of the linearized homogenization decreases. For example, setting $k = 0.1$, as in Figure 3.5c, results in an error on the order of 0.01. In all cases, the error is near zero in a large part of the domain. It concentrates near the positive diagonal, where it is .1 to .4 for the non-smooth operator, and similar for the operator with a small smoothing parameter. A larger smoothing parameter sends the error in a similar region to the range .002 to 0.01. A small amount of smoothing has a small effect on the error. More smoothing leads to errors going from .1 to .002 in a similar part of the domain.



a $a_0(y)F^{3,1}(Q)$, $r = 2$ **b** $a_0(y)F^{3,1}_{10}(Q)$, $r = 2$ **c** $a_0(y)F^{3,1}_{0.1}(Q)$, $r = 2$

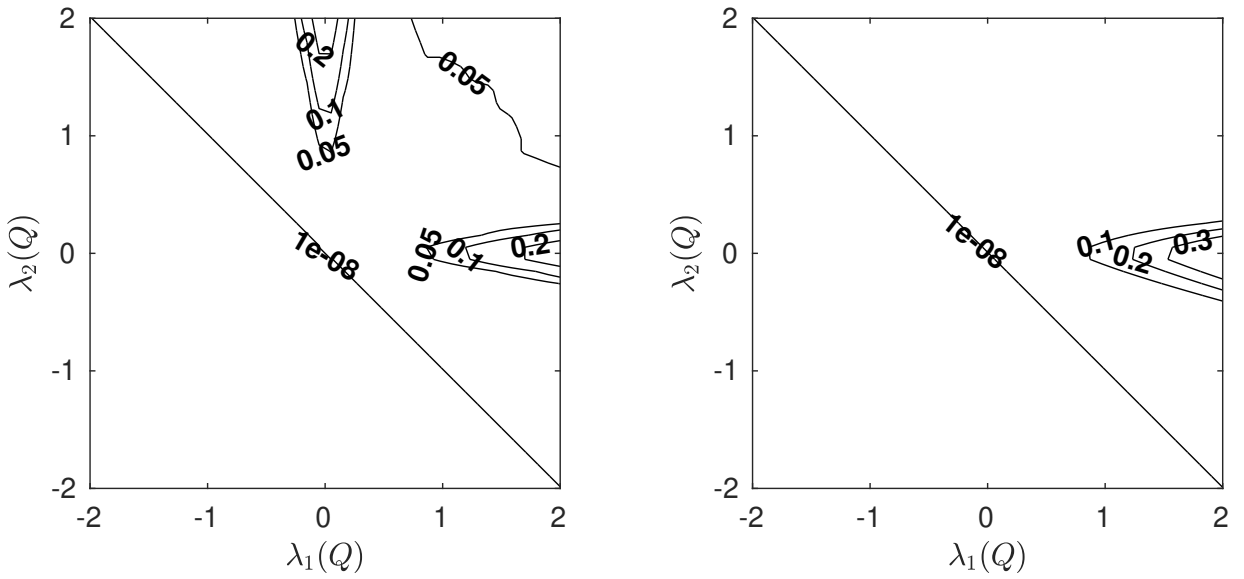
Figure 3.5: Homogenization error for a smoothed Pucci type operator. The coefficients $a(y)$ are on a checker board with $r = 2$ (i.e. a = 1 or 2). The operators are defined in Section 3.4. Figure 3.5a: error on a Pucci like operator. Figures 3.5b and 3.5c: error on a smoothed Pucci like operator.



a $r = 1.2$ **b** $r = 2$

Figure 3.6: Error for $a_0(y)F^{3,1}$ on stripes, with different ratios r.

Figure 3.6 presents the error for $a_0(y)F^{3,1}(Q)$ on stripes. On stripes, the regions with large error are much smaller than the operators on checkerboard. We hypothesize that this is because stripes have a smoothing effect. The location where the large error is located depends on the interplay between the operator and the direction of the stripes. Given that in this example the homogenized operator is $\mathcal{O}(1)$, the error here is particularly large. In a companion paper, we will derive a closer lower bound for $\overline{F^{A,a}}(Q)$, using the optimal invariant measure of the nonlinear operator.



a *Checkerboard* **b** *Stripes*

Figure 3.7: Error for $M(Q, y)$ with $r = 2$, on a periodic checkerboard and on stripes.

Figure 3.7 shows error for $a_0(y)(\mathrm{Tr}(Q)+\mathrm{MA}(Q))$ on both stripes and a periodic checkerboard. For the Monge Ampere type operator on checkerboard, error is on the order of $1\mathrm{e}{-2}$ in the first quadrant, where the curvature is bounded. Elsewhere the error is negligible.

As r (the scaling coefficient of $a_0(y)$) grows, so does the error. As expected, for the two Pucci type operators on checkerboard, away from the regions where the curvature is unbounded, the error is negligible: this is where the operators are linear. Although we do not show it, for all figures, the error profile on the random checkerboard is nearly identical to the periodic checkerboard.

3.4.2 *Numerical Results: non-separable operators*

Now we consider nonseparable coefficients for $F^{A,a}(Q, y)$, refer to Definition 3.3.1.

For both periodic and random checkerboard coefficients, the numerically computed values of $\overline{F^{A,a}}(Q)$ depend only on the eigenvalues of Q, not on the eigenvectors. In addition, $\overline{F^{A,a}}(Q)$ is homogeneous order one. So the entire function $\overline{F^{A,a}}(Q)$ is determined by the 1-level set of $\overline{F^{A,a}}(Q)$ for diagonal matrices Q.

We write

$$\overline{L^Q}(Q) = \overline{A}\lambda^+(Q) + \overline{a}\lambda^-(Q) \tag{3.26}$$

where the coefficients are obtained by numerical homogenization of the linearized operator (3.3) when Q had at least one positive eigenvalue. (In the negative definite case the operator

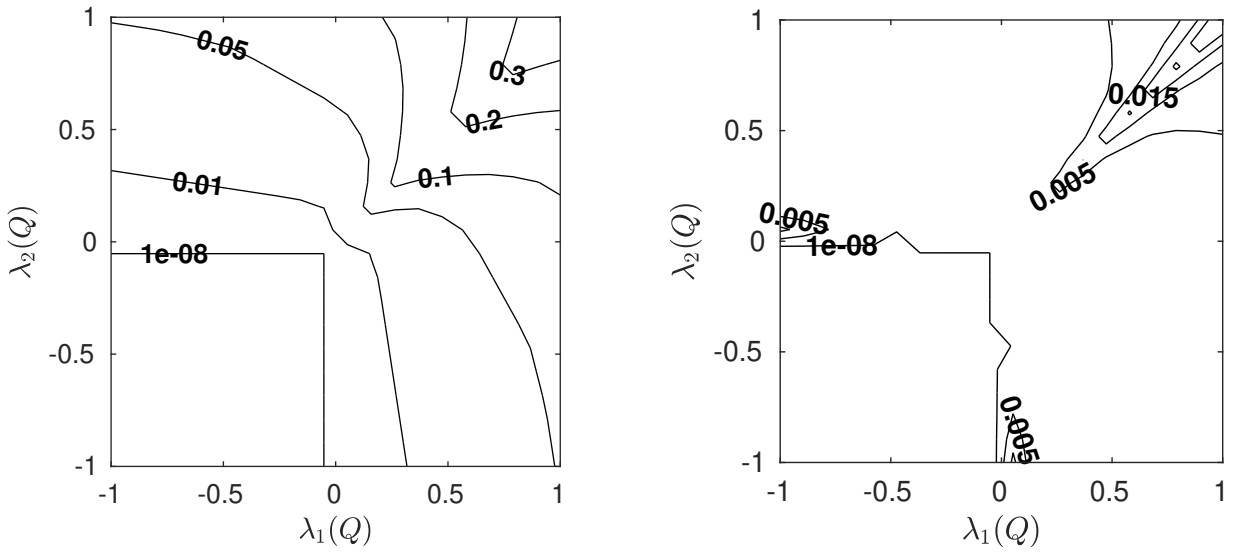


Figure 3.8: Error for the non-separable operator on a periodic checkerboard. Figure 3.8a: alternating between $F^{1,1}$ and $F^{4,1}$. Figure 3.8b: alternating between $F^{2,1}$ and $F^{1,\frac{4}{3}}$.

is linear and the error was insignificant).

We found that error was within 5% for a range of values of A and a with coefficients which vary by a factor of 10.

In Figure 3.8 we show the error on a periodic checkerboard, with

$$F^{A,a}(Q,y) = \begin{cases} \mathrm{Tr}(Q),\ y \in B \\ F^{4,1},\ y \in W. \end{cases}$$

The error is on the order of $1\mathrm{e}{-1}$ near the line $\lambda^+ = \lambda^-$ in the first quadrant; on the order of $1\mathrm{e}{-2}$ in the second and fourth quadrants; and negligible otherwise. In Figure 3.8 we plot the error against the numerically homogenized value for an the nonconvex operator alternating between $F^{2,1}$ and $F^{1,\frac{4}{3}}$ on a periodic checkerboard.

3.4.2.1 *Further experiments*

We let A and a each take two positive values in periodic checkerboard pattern. In the second, we let A and a each take two positive values in a random checkerboard, drawn randomly from a Bernoulli trial with probability p. We checked both when $p = \frac{1}{2}$ and other values of p. When $p = \frac{1}{2}$ the homogenized operator on the random checkerboard is identical to the homogenization on the periodic checkerboard. Finally, we also checked the case when A and a are each drawn from a uniform distribution with positive support.

In all of these cases, the numerically homogenized operator is (numerically) isotropic, homogeneous order one, and agrees closely with $\overline{F}$ in the approximate formula (3.26).

3.5 CONCLUSIONS

We studied the error between the homogenization of the linearized operator and the fully nonlinear homogenization. We obtained upper and lower bounds on the error in terms of the generalized semiconvavity constants of the operator.

We also performed numerical calculations. For the class of operators we studied, linearization was very accurate for a wide range of values of Q, with negligible error in some cases. The numerically computed errors were small, and concentrated around regions of high curvature in Q of the operator $F(Q, x)$. Errors grew with the degree of nonlinearity and with the range of the coefficients.

The numerical results are consistent with the bounds, although in some cases the error was smaller than was predicted by the bounds.

CHAPTER 4

IMPROVED ACCURACY OF MONOTONE FINITE DIFFERENCE SCHEMES ON POINT CLOUDS AND REGULAR GRIDS

Abstract

Finite difference schemes are the method of choice for solving nonlinear, degenerate elliptic PDEs, because the Barles-Sougandis convergence framework [BS91] provides sufficient conditions for convergence to the unique viscosity solution [CIL92]. For anisotropic operators, such as the Monge-Ampere equation, wide stencil schemes are needed [Obe06]. 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 [Fro18]. 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.

4.1 INTRODUCTION

The goal of this paper is to build more accurate convergent discretizations for the class of nonlinear elliptic partial differential equations [CIL92]. Our schemes are applicable in both two and three dimensions for a class of PDEs, which include the convex envelope operator and the Pucci operator, as well as the Monge-Ampere 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, [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 non-regular 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-Ampere 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.

Higher accuracy is also possible using filtered schemes [FO13, OS15, BPR16] Filtered schemes combine a base monotone scheme with a higher accuracy schemes: 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] and [CW17c], where the Monge-Ampere 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 [CW17c], 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 [NNZ19] prove 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 arise 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 $\mathcal{O}(R + d\theta)$ meshfree 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, §5.4 p.595] to write down the discretization. For first order derivates, 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.

4.1.1 *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 $d\theta = \arccos\langle w, v_h/\|v_h\|\rangle$. 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 R is $\mathcal{O}(\frac{h}{d\theta})$ (see for example [Fro18] for details). With optimal choice $d\theta = (2h^2)^{\frac{1}{3}}$, 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 $\mathcal{O}\left(\sqrt{h}\right)$. 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 4.2.2), the scheme can easily be implemented near the boundary of a domain as well, with an appropriate choice of

Table 4.1: 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, non symmetric	$\mathcal{O}(R + d\theta^2)$	$\mathcal{O}(h^{\frac{1}{3}})$	$\mathcal{O}(h^{\frac{2}{3}})$	n-d, mesh free. No difficulty at boundary.

boundary resolution (see Remark 4.1). In particular, the scheme easily handles Neumann boundary conditions on non rectangular domains.

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

Table 4.1 presents a summary of the second derivative schemes discussed in this paper.

4.1.2 *Directional discretizations*

The basic building block 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

$$\mathcal{D}_{tw}u(x) = u_w(x) + \mathcal{O}(t) \tag{4.1}$$
$$\mathcal{D}_{-tw}u(x) = u_w(x) + \mathcal{O}(t)$$

The simplest finite difference scheme for u_{ww} is the centred finite differences

$$\frac{u(x+tw) - 2u(x) + u(x-tw)}{t^2} = \frac{1}{t}\left[\mathcal{D}_{tw}u(x) + \mathcal{D}_{-tw}u(x)\right] \tag{4.2}$$
$$= u_{ww}(x) + \mathcal{O}(t^2) \tag{4.3}$$

The generalization to unequally spaced points is clear from (4.2)

$$\frac{2}{t_p + t_m}\left[\mathcal{D}_{t_p w}u(x) + \mathcal{D}_{-t_m w}u(x)\right] = u_{ww}(x) + \mathcal{O}(t_+). \tag{4.4}$$

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

4.1.3 *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, §5.4 p.595]), which allows us to generalize (4.2).

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

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

Suppose further that

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

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

$$L_p(x) = \sum_{i \in \mathcal{S}_p} \lambda_p^i(x) u(x_i) \tag{4.5}$$
$$L_m(x) = \sum_{i \in \mathcal{S}_m} \lambda_m^i(x) u(x_i). \tag{4.6}$$

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 $r \leq ||v_i|| \leq R$. Let V_p be the matrix

$$V_p = \begin{bmatrix} v_1^p & v_2^p & \dots & v_n^p \end{bmatrix}. \tag{4.7}$$

Then λ_p is given by solving

$$V_p \lambda_p = x. \tag{4.8}$$

The barycentric coordinates λ_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 4.1.1 (First derivative schemes)**.** *The first derivative scheme takes two forms, respectively upwind and downwind:*

$$\mathcal{D}_w u(x_0) := \frac{1}{t_p} \left(L_p(x_0 + t_p w) - u(x_0) \right), \qquad t_p = \frac{1}{1^\mathsf{T} V_p^{-1} w} \tag{4.9}$$

$$\mathcal{D}_{-w} u(x_0) := \frac{1}{t_m} \left(L_p(x_0 - t_m w) - u(x_0) \right), \qquad t_m = \frac{-1}{1^\mathsf{T} V_m^{-1} w} \tag{4.10}$$

Definition 4.1.2 (Second derivative scheme)**.** *The second derivative scheme is defined as*

$$\mathcal{D}_{ww} u(x_0) = \frac{2 \left(\mathcal{D}_w u(x_0) + \mathcal{D}_{-w} u(x_0) \right)}{t_p + t_m}. \tag{4.11}$$

with t_p and t_m given above.

Lemma 4.1.3 (Monotone and stable)**.** *The finite difference schemes of Definitions 4.1.1 and 4.1.2 are monotone and stable.*

Proof. By convexity, we are guaranteed that $0 \leq \lambda_{p,m}^i \leq 1$. Further, we have that both $\sum \lambda_p^i = \sum \lambda_m^i = 1$. 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.

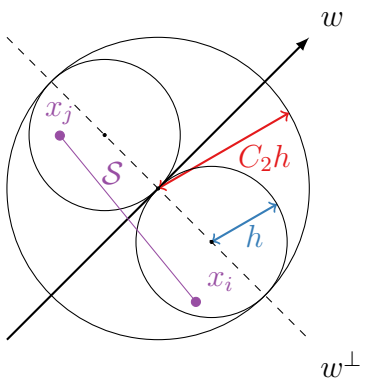
4.2 THE FRAMEWORK

In this section we introduce a framework for constructing monotone finite difference operators on a point cloud, in dimensions two or three. To implement the method, we require finding triangles (in two dimensions) or tetrahedra (in three dimensions) which contain the reference point. The neighbours 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 §4.2.3 for details.) The configuration of these simplices determines the accuracy of the scheme.

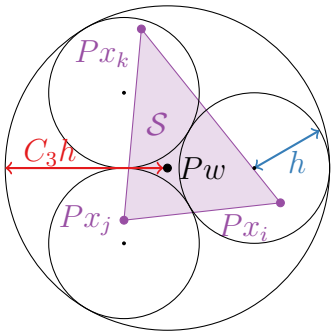
4.2.1 *Notation*

We use the following notation.

- $\Omega \subset \mathbb{R}^n$, 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.
- $h = \sup_{x\in\Omega} \min_{y\in\mathcal{G}} \|x - y\|$, 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 minimum distance between an interior point and a boundary point.
- ℓ is the minmum length of all edges in the graph $\mathcal{G}$.
- $d\theta$ is the desired angular resolution. We shall require at least $d\theta < \pi$.
- $R = C_n h \left(1 + \operatorname{cosec}(\frac{d\theta}{2})\right)$ is the maximal search radius, and depends only on the angular resolution, the spatial resolution, and a constant C_n determined by the dimension.
- $r = C_n h \left(-1 + \operatorname{cosec}(\frac{d\theta}{2})\right)$ 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 $\delta \geq r$.



a $C_2 = 2$



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

Figure 4.1: There exists an $n-1$ simplex $\mathcal{S}$ enclosing w, contained within ball of radius $C_n h$. In Fig 4.1b, projections onto a plane perpendicular to w are shown.

- 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 require the existence of simplices which intersect the vector w. For accuracy, we further require that 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. Refer to Figure 4.1.

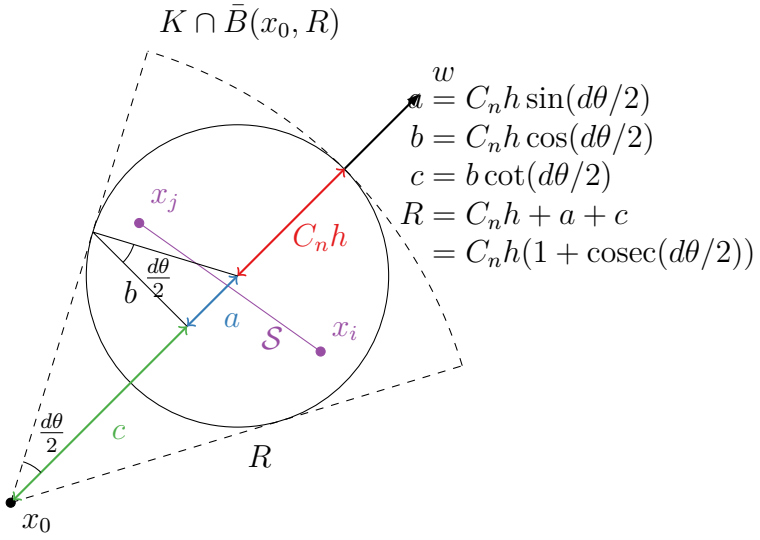
Lemma 4.2.1 (Existence of scheme away from boundary)**.** *Take $x_0 \in \mathcal{G}$ with $\operatorname{dist}(x_0, \partial\Omega) \geq R$. Then it is possible to construct the simplices used in Definition 4.1.1.*

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

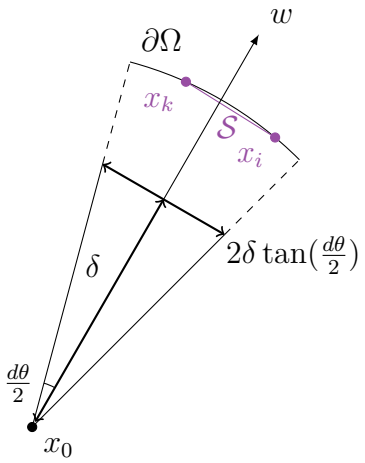
$$K := \left\{ x \mid \frac{\langle v, w \rangle}{\|v\|} \geq 1 - \cos(\frac{d\theta}{2}), v = x - x_0 \right\}. \tag{4.12}$$

Any two points in K have 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 Ω 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' problem), it can be shown that these n balls of radius h are all contained within a larger ball of radius $C_n h$ (with $C_2 = 2$ and $C_3 = 1 + \frac{2}{\sqrt{3}}$).



a *Interior simplex*



b *Boundary simplex*

Figure 4.2: Construction of simplices for the finite difference scheme on: (4.2a) an interior point sufficiently far from the boundary; and (4.2b) a point near the boundary.

Refer to Figure 4.1. Thus, a candidate simplex is guaranteed to exist within every ball of radius $C_n h$ with 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 4.2a. 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 4.2.2 (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 schemes given by Definition 4.1.1 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 4.2.1.

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

$$\|x_0 - y\| < R \text{ if } y \in \partial\Omega \cap K. \tag{4.13}$$

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 similar geometrical reasoning as in the previous lemma (see Figure 4.2b), 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 δ. For our purposes, we guarantee the existence of a scheme for the normal derivative with the following lemma.

Lemma 4.2.3 (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 Ω_δ 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 δ 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 Ω. Thus, by the same arguments as in the proof of Lemma 4.2.1, the simplex $\mathcal{S}_p$ exists and has angular resolution less than $d\theta$. This allows for the construction of (4.9) for the normal derivative. □

Combining these three lemmas guarantees existence of the schemes.

Theorem 4.2.4 (Existence of schemes)**.** *Suppose $\mathcal{G}$ is a point cloud in Ω 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 4.1.1 and 4.1.2 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 4.1 (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 4.2.4 with optimal choice $d\theta \approx h^\alpha$ gives that $h_B = \mathcal{O}(h^{1+\alpha})$. The constant α is $\frac{1}{2}$ for regular grids and $\frac{1}{3}$ for point clouds, see Table 4.1.

4.2.2 *Consistency & 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

$$u(x) - L(x) \approx \frac{1}{2} \sum \lambda_j(x)(x - x_j)^\mathsf{T} D^2 u(x_j)(x - x_j). \tag{4.14}$$

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

$$\begin{aligned} E[L_p] :=& u(x_0 + t_p w) - L_p(x_0 + t_p w) && (4.15)\\ \approx& \frac{1}{2} \sum_{i \in \mathcal{S}_p} \lambda_p^i \left(v_i^p - t_p w\right)^\mathsf{T} D^2 u(x_i) \left(v_i^p - t_p w\right) && (4.16)\\ \leq& \frac{1}{2} ||D^2 u||_\infty \sum_{i \in \mathcal{S}_p} \lambda_p^i ||v_i^p - t_p w||^2. && (4.17) \end{aligned}$$

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

Lemma 4.2.5 (Consistency of first derivative scheme)**.** *The first derivative schemes of Definition 4.1.1 are consistent with a formal discretization error of $\mathcal{O}(h)$.*

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

$$\begin{aligned} E[\mathcal{D}_w u, d\theta] =& \frac{E[L_p]}{t_p} && (4.18)\\ \leq& \frac{1}{2} \|D^2 u\|_\infty \sum_{i \in \mathcal{S}_p} \lambda_i^p \frac{\|v_i^p - t_p w\|^2}{t_p} && (4.19)\\ \leq& \frac{1}{2} \|D^2 u\|_\infty \frac{\max_{i,j \in \mathcal{S}_p} \|v_i^p - v_j^p\|^2}{\min_{k \in \mathcal{S}_p} \|v_k\|}. && (4.20) \end{aligned}$$

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

$$\min_{k\in\mathcal{S}_p,\mathcal{S}_m} \|v_k\| \geq r = C_n h\left(-1+\operatorname{cosec}(\frac{d\theta}{2})\right) \tag{4.21}$$

$$= \mathcal{O}\left(\frac{h}{d\theta}\right). \tag{4.22}$$

With this in mind, (4.20) is bounded by

$$E[\mathcal{D}_w u, d\theta] \leq \frac{1}{2}\|D^2 u\|_\infty \frac{(2C_n h)^2}{r} \tag{4.23}$$

$$= \mathcal{O}(h d\theta) \tag{4.24}$$

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

Lemma 4.2.6 (Consistency of second derivative schemes)**.** *Using a non symmetric 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 4.1.2 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

$$E[\mathcal{D}_{ww}u, d\theta] = 2\left(\frac{E[L_p]}{t_p^2 + t_p t_m} + \frac{E[L_m]}{t_m^2 + t_p t_m}\right) \tag{4.25}$$

$$\leq \frac{1}{t_-^2}\Big(E[L_p] + E[L_m]\Big) \tag{4.26}$$

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

$$E[\mathcal{D}_{ww}u, d\theta] \leq ||D^2 u||_\infty \frac{\max_{S\in\mathcal{S}_p,\mathcal{S}_m} \max_{i,j\in S} ||v_i - v_j||^2}{\min_{k\in\mathcal{S}_p,\mathcal{S}_m} ||v_k||^2} \tag{4.27}$$

$$\leq ||D^2 u||_\infty \frac{(2C_n h)^2}{r^2} \tag{4.28}$$

$$= \mathcal{O}(d\theta^2). \tag{4.29}$$

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; the angular error from the linear interpolation). For the second derivative, in the non symmetric case, the error of the scheme is

$$E[u_{ww}] = \mathcal{O}(R + d\theta^2) \tag{4.30}$$

$$= \mathcal{O}(\frac{h}{d\theta} + d\theta^2), \tag{4.31}$$

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

$$E[\mathcal{D}_{ww}u] = \mathcal{O}(R^2 + d\theta^2) \tag{4.32}$$

$$= \mathcal{O}\left(\left(\frac{h}{d\theta}\right)^2 + d\theta^2\right) \tag{4.33}$$

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 non symmetric 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 4.2. 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 4.3. 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, $\delta \geq r$.

4.2.3 *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 $\mathcal{S}_k \in \mathcal{L}_i$, $k = 1, \ldots, m_i$, 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 neighbour points, denoted by V, is unique. If not, for each set of non unique 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

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

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

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

Then by Farkas' 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 4.4. The proofs of Section 4.2 relied on choosing the maximal and minimal search radii to respectively be $R, r = C_n h(\pm 1 + \operatorname{cosec}(\frac{d\theta}{2}))$. 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.

4.3 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 non empty. For each pair $\{\mathcal{S}_p, \mathcal{S}_m\}$ of overlapping antipodal simplices in $\mathcal{L}_i$ (with $K(\mathcal{S}_p) \cap K(-\mathcal{S}_m) \neq \emptyset$),

[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 $\mathcal{L}_i = \{\mathcal{S}_1, \ldots, \mathcal{S}_{m_i}\}$

$\mathcal{T} \leftarrow$ triangulation$(\mathcal{G})$; // triangulation of $\mathcal{G}$
$A \leftarrow$ 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
$R \leftarrow C_n h \left(1 + \text{cosec}\left(\frac{d\theta}{2}\right)\right)$; // maximum search radius
$r \leftarrow C_n h \left(-1 + \text{cosec}\left(\frac{d\theta}{2}\right)\right)$; // minimum search radius
$p \leftarrow \lceil p(R) \rceil$; // maximum neighbour graph distance
$P \leftarrow \sum_{k=1}^{p} A^k$
foreach $i \in \mathcal{I}$ **do**
  $\mathcal{N} \leftarrow \{j \mid P_{ij} \neq 0,\ i \neq j,\ r \leq \|x_i - x_j\| \leq R\}$; // Neighbour 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 $\{\mathcal{L}_i\}$, $i \in \mathcal{I}$

one computes

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

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 the Remark 4.5.)

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

antipodal pairs:

$$\Lambda_+^{h,d\theta} u(x_i) = \max_{\substack{\mathcal{S}_m, \mathcal{S}_p \in \mathcal{L}_i \\ K(\mathcal{S}_m) \cap K(-\mathcal{S}_p) \neq \emptyset}} P[\mathcal{S}_m, \mathcal{S}_p] \tag{4.34}$$

The error of the scheme is

$$E[\Lambda_+^{h,d\theta}] = \left| \max_{\|v\|=1} v^\mathsf{T} D^2 u(x_i) v - \max_{\|w\|=1} \mathcal{D}_{ww} u(x_i) \right| \tag{4.35}$$

$$\leq \max_{\|w\|=1} w^\mathsf{T} D^2 u(x_i) w - \mathcal{D}_{ww} u(x_i) \tag{4.36}$$

$$= \mathcal{O}(R + d\theta^2), \tag{4.37}$$

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

Remark 4.5. In cases other than on a regular grid in two dimensions, the optimization problem (4.34) 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

$$\cos d\theta_e = \max_i \min_{j \neq i} \langle w_i, w_j \rangle. \tag{4.38}$$

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

$$\Lambda_+^{h,d\theta_e} u(x_i) = \max_i \mathcal{D}_{w_i w_i} u(x_i). \tag{4.39}$$

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

4.4 SOLVERS

Before continuing with specific numerical examples, we first detail the numerical solver used. All solutions in Section 4.5 were computed with a global semi-smooth 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 semi-smooth 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 $\|F^h[u]\|^2$. 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 one Newton step, which was first proposed in [Car17]. Because the Euler step is a guaranteed descent direction, the method is globally convergent [FP07].

4.5 Numerical Examples

Here we test our meshfree 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]).

4.5.1 *Convex envelope*

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

$$\begin{cases} \max\{u(x) - g(x), -\Lambda_- u(x)\} = 0 & x \in \Omega \\ u(x) = g(x) & x \in \partial\Omega, \end{cases} \tag{4.40}$$

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,

$$g(x) = \min_{i=1,2}\{\|x - p_i\|\}, \tag{4.41}$$

or in otherwords, 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 4.3, and using the wide stencil method developed in [Obe08a]. We call the latter a nearest neighbour scheme. For both methods, we solved the equation using stencils with radius two and three. Figure 4.3b and Table 4.2 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.

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

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 neighbour method, for the values of h we studied, the linear interpolation 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 neighbours. Thus the spatial resolution error of the linear interpolation scheme is generally higher than the nearest neighbour 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 grid has been rotated off axis. Our results are presented in Figure 4.4. The mean of the error of the linear interpolation scheme is higher than the nearest neighbour scheme, due to the fact that the linear interpolation scheme chooses points further from the stencil centre. However, the variance of the error for the linear interpolation scheme nearest neighbour scheme is much less than that of the nearest neighbour 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 4.3a and Table 4.2. We can see that the linear interpolation scheme achieves both the best rate of convergence and a better absolute error.

4.5.2 *Pucci equation*

Our next example is the Pucci equation,

$$\begin{cases} \alpha\Lambda_+ u(x) + \Lambda_- u(x) = 0 & x \in \Omega \\ u(x) = g(x) & x \in \partial\Omega \end{cases} \tag{4.42}$$

where α 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

$$u(x,y) = -\rho^{1-\alpha}, \quad \rho(x,y) = \sqrt{(x+2)^2 + (y+2)^2}. \tag{4.43}$$

Table 4.2: 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 neighbour, $r = 2$			
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 neighbour, $r = 3$			
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

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

We discretized the square using a regular grid, and use either the nearest neighbour scheme, or the symmetric finite difference interpolation scheme presented in Section 4.3. We use stencils of radius two or three. Errors and rates of convergence on the grid are presented in Table 4.3 and in Figure 4.5b. Both methods achieve roughly the same convergence rate before angular resolution error dominates. The nearest neighbour 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 4.3 and Figure 4.5a. 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.

4.5.2.1 *Solver comparison*

Finally, we performed a comparison of the three solvers (semi-smooth 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 4.4.

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 neighbour. For the interpolation finite different schemes, both semi-smooth Newton and the combination solver is nearly $\mathcal{O}(N)$: we calculated a log-log line of best fit, and found semi-smooth Newton and the combination solver to be about $\mathcal{O}(N^{1.2})$.

Of all solvers and finite difference methods, the nearest neighbour finite difference scheme with the combination solver achieves the best CPU time, followed by the semi-smooth 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 different scheme, and so we expect on even larger grids, eventually the interpolation finite difference method would be faster with either semi-smooth Newton or the combination solver.

Table 4.3: 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 neighbour, $r = 2$			
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 neighbour, $r = 3$			
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

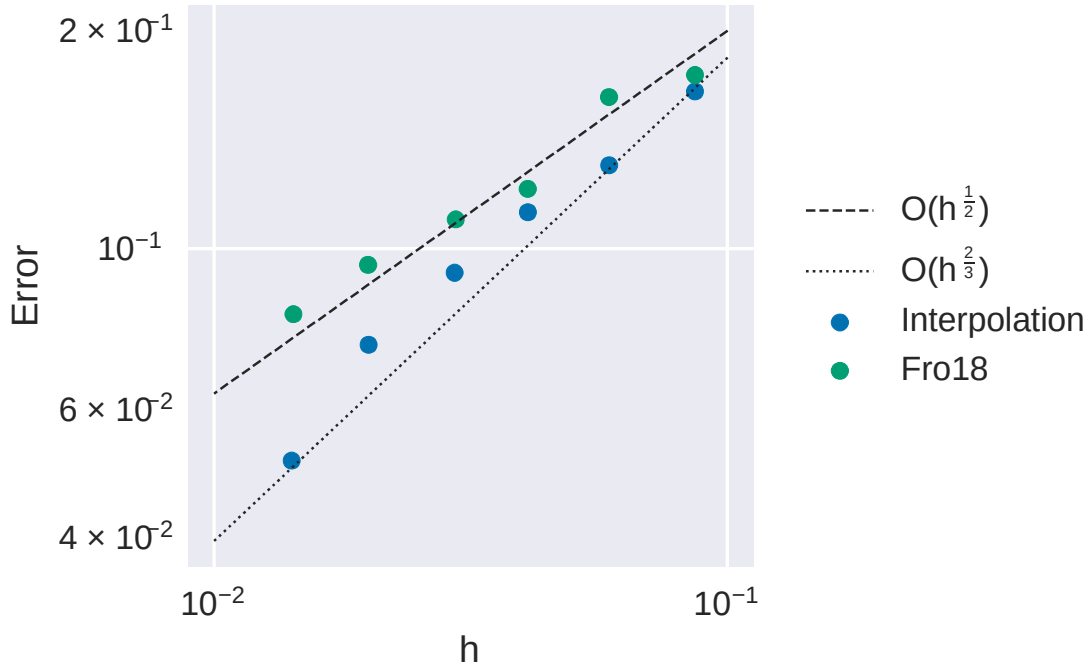
Table 4.4: Comparison of wall clock time of solvers for the Pucci equation (4.42) in two dimensions on a regular grid. Time is reported in seconds. Results are for stencils of either radius $r = 2$ or $r = 3$.

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

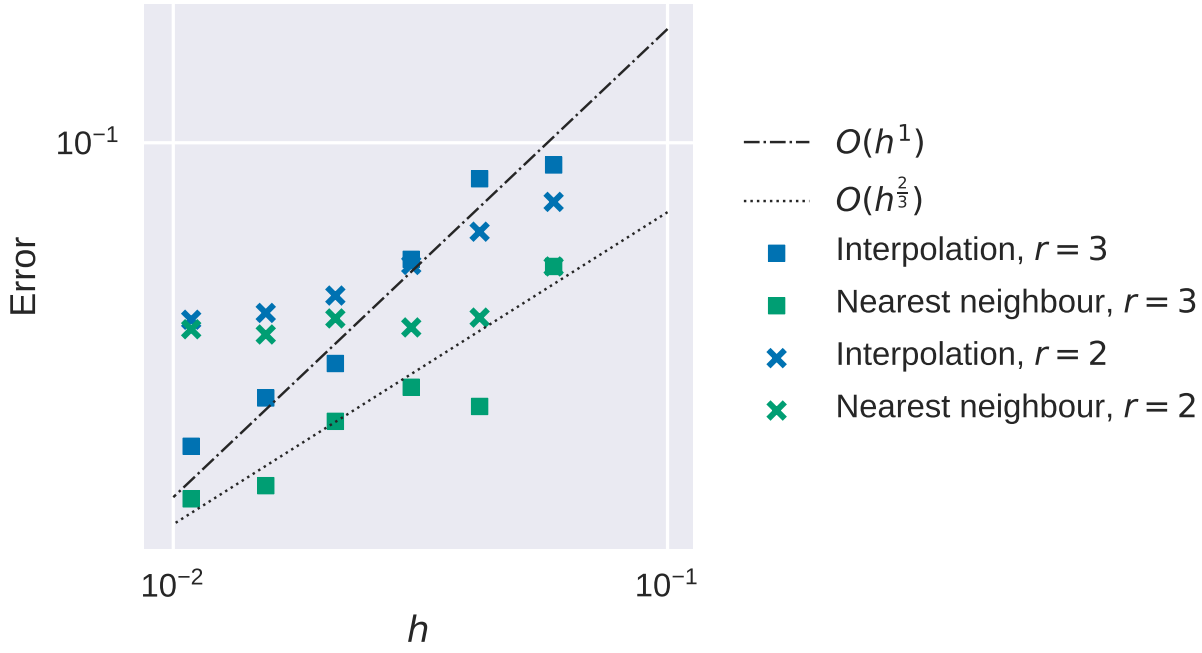
Nearest neighbour, $r = 2$						
N	392	721	1288	2492	4616	9017
Euler	0.43	1.62	5.75	24.23	90.90	383.43
Newton	0.06	0.11	0.25	1.04	3.40	13.51
Combination	0.05	0.10	0.23	0.73	2.66	9.54

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 neighbour, $r = 3$						
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*

Figure 4.3: Figure 4.3a: Convergence plot for the convex envelope on the unit disc with triangular mesh. Figure 4.3b: Convergence plot for the convex envelope on a regular grid over the square $[-1, 1]^2$.

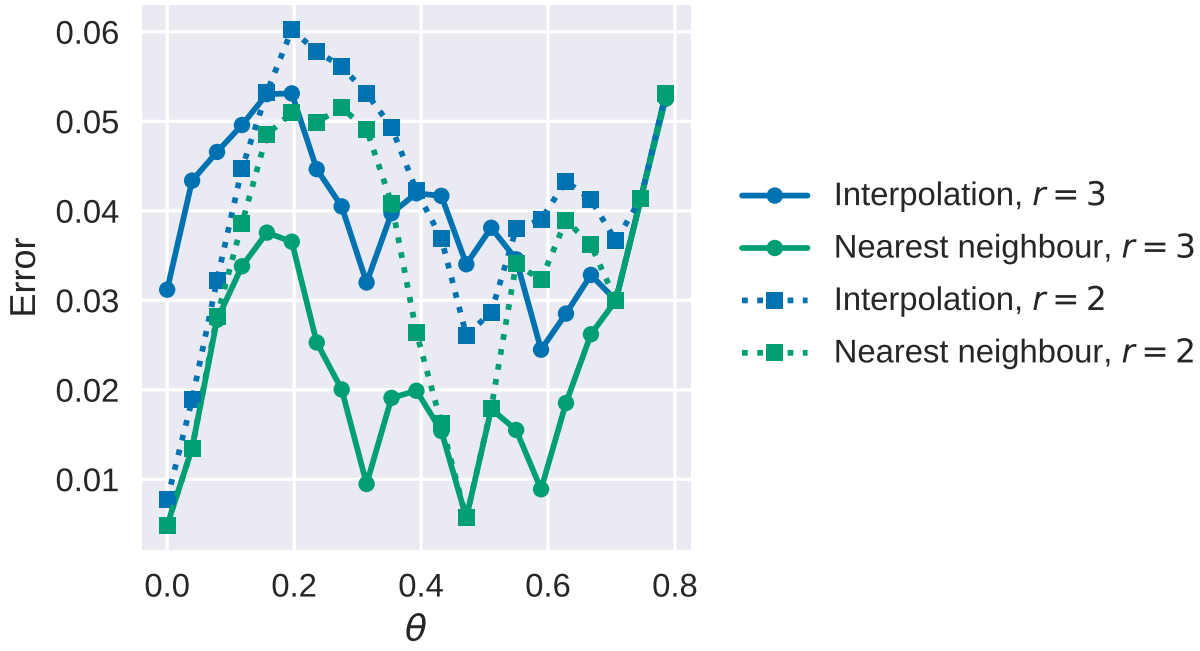
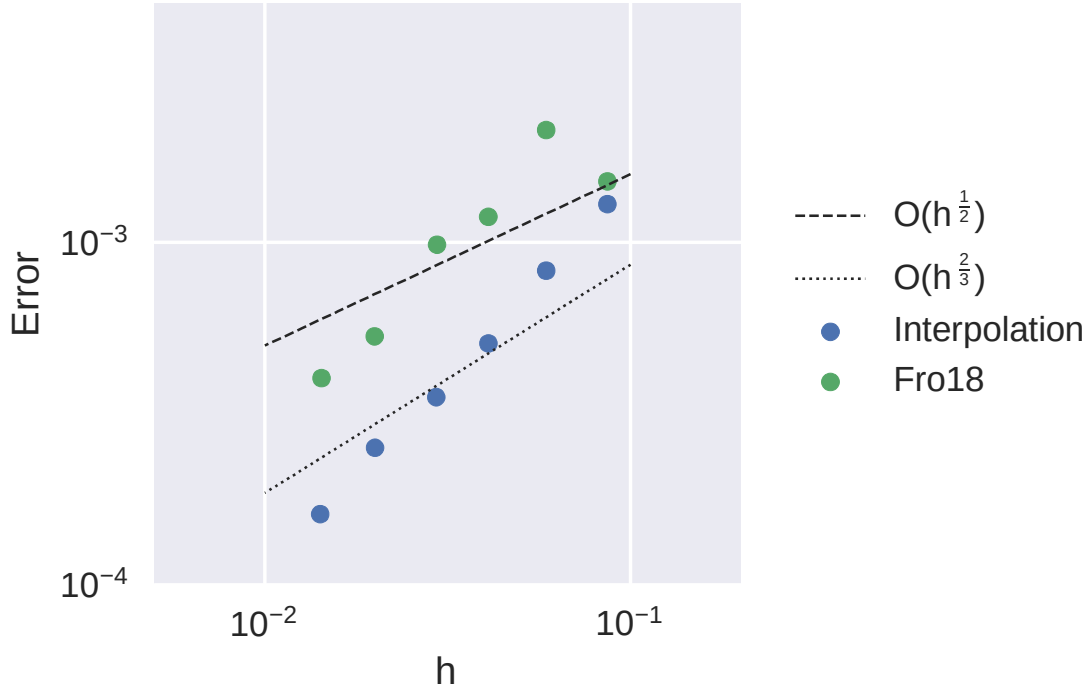
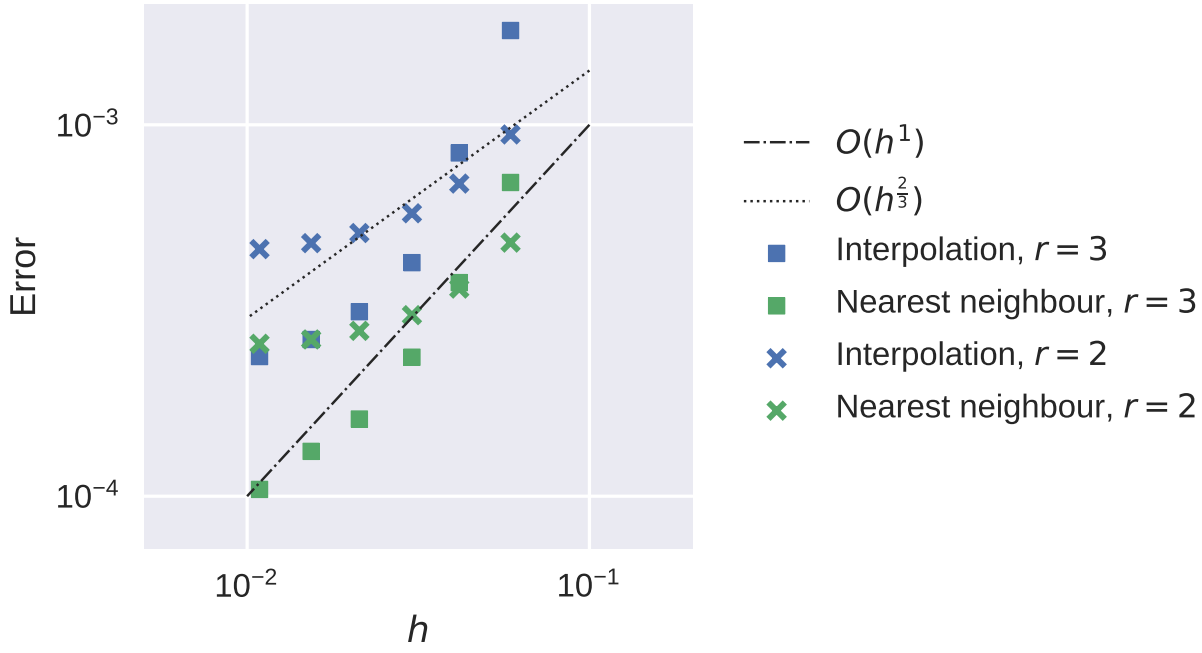


Figure 4.4: Error of the numerical solutions of the convex envelope PDE on a regular grid, as a function of rotation of the grid.

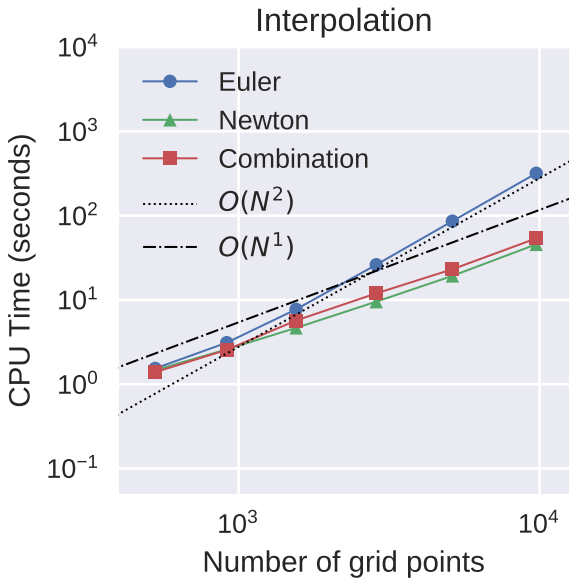


a *Triangular mesh*

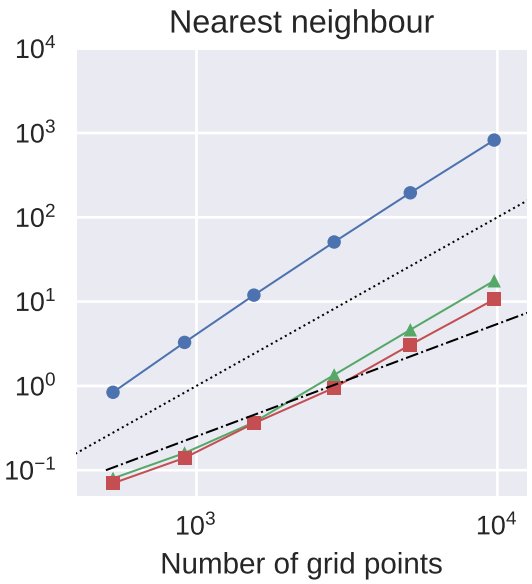


b *Regular grid*

Figure 4.5: Figure 4.5a: Convergence plot for the Pucci equation on the unit disc with triangular mesh. Figure 4.5b: Convergence plot for the Pucci equation on a regular grid over the square $[-1, 1]^2$.



a *Triangular mesh*



b *Regular grid*

Figure 4.6: CPU time taken to compute solution of the Pucci equation on a regular grid, with stencil width $r = 3$, for both methods.

CHAPTER 5

SCALEABLE INPUT GRADIENT REGULARIZATION FOR ADVERSARIAL ROBUSTNESS

Abstract

Input gradient regularization is not thought to be an effective means for promoting adversarial robustness. In this work we revisit this regularization scheme with some new ingredients. First, we derive new per-image theoretical robustness bounds based on local gradient information, and curvature information when available. These bounds strongly motivate input gradient regularization. Second, we implement a scaleable version of input gradient regularization which avoids double backpropagation: adversarially robust ImageNet models are trained in 33 hours on four consumer grade GPUs. Finally, we show experimentally that input gradient regularization is competitive with adversarial training.

5.1 INTRODUCTION

Neural networks are vulnerable to *adversarial attacks*. These are small (imperceptible to the human eye) perturbations of an image which cause a network to misclassify the image [BCM+13, SZS+13, GSS14]. The threat posed by adversarial attacks must be addressed before these methods can be deployed in error-sensitive and security-based applications [Pot17].

Building adversarially robust models is an optimization problem with two objectives: (i) maintain test accuracy on clean unperturbed images, and (ii) be robust to large adversarial perturbations. The present state-of-the-art method for adversarial defence, adversarial training [SZS+13, GSS14, TKP+18, MMS+17, MMIK18], in which models are trained on perturbed images, offers robustness at the expense of test accuracy [TSE+18]. It is not clear that multi-step adversarial training is scaleable to large datasets such as ImageNet-1k [DDS+09]. Previous attempts [KKG18, XWvdM+18] used hundreds of GPUs and took nearly a week to train, although recent work by Shafahi et al. [SNG+19] has offered a remedy.

Assessing the *empirical* effectiveness of an adversarial defence requires careful testing with multiple attacks [GMP18]. Furthermore, existing defences are vulnerable to new, stronger attacks: two recent works [CW17a, ACW18] have advocated designing specialized

attacks to circumvent prior defences, while Uesato et al. [UOKvdO18] warn against using weak attacks to evaluate robustness. This has led the community to develop *theoretical* tools to certify adversarial robustness. Several certification approaches have been proposed: through linear programming [WK18, WSMK18] or mixed-integer linear-programming [XTSM18]; semi-definite relaxation [RSL18b, RSL18a]; randomized smoothing [LCWC18, CRK19]; or estimates of the local Lipschitz constant [HA17, WZC+18, TSS18]. The latter two approaches have scaled well to ImageNet-1k.

In practice, certifiably robust networks often perform worse than adversarially trained models, which lack theoretical guarantees. In this article, we work towards bridging the gap between theoretically robust networks and empirically effective training methods. Our approach relies on minimizing a loss regularized against large input gradients

$$\mathop{\mathbb{E}}_{(x,y)\sim\mathbb{P}}\left[\mathcal{L}(f(x;w),y)+\frac{\lambda}{2}\|\nabla_x\mathcal{L}(f(x;w),y)\|_*^2\right] \tag{5.1}$$

where $\|\cdot\|_*$ is dual to the one measuring adversarial attacks (for example the ℓ_1 norm for attacks measured in the ℓ_∞ norm). Heuristically, making loss gradients small should make gradient based attacks more challenging.

Drucker and LeCun [DL91] implemented gradient regularization using 'double back-propagation', which has been shown to improve model generalization [NBA+18]. It has been used to improve the stability of GANs [RLNH17, NK17] and to promote learning robust features with contractive auto-encoders [RVM+11]. While it has been proposed for adversarial attacks robustness [RD18, RLNH18, HA17, JG18, SOS+18], experimental evidence has been mixed, in particular, input gradient regularization has so far not been competitive with multi-step adversarial training.

On non-smooth networks (such as those built of ReLUs) small gradients are no guarantee of adversarial robustness [PMG+17], and so it is thought input gradient regularization should not be effective on non-smooth networks. This raises the question, how often is the lack of smoothness an issue, in practice? In other words, when do Taylor approximations of the loss fail to predict adversarial robustness, and is smoothness only needed theoretically? The fact that first-order gradient-based attacks of the loss (like PGD [MMS+17]) are usually effective indicates that in many scenarios, non-smoothness is not an issue. However in a non-negligible minority of cases, attacks based on decision boundary information [CW17b, BRB18, CJ19, FPO19] outperform gradient based attacks. This indicates the curvature near these points is large, and first-order information is not sufficient to guarantee robustness. We illustrate this point in Fig 5.1. In this work we overcome the limitation of gradient regularization for non-smooth networks by instead building networks of 'smooth'

ReLUs. At the expense of a minor drop in test accuracy, we obtain tighter theoretical lower bounds on robustness, since we can better approximate the loss using local information.

Another drawback of input gradient regularization is that it is not presently tractable to update model weights using double backpropagation on large networks. We circumvent this limitation by differentiating the regularization term without double backpropagation.

Our main contributions are the following. First, we motivate using input gradient regularization *of the loss* by deriving new theoretical robustness bounds. These bounds show that small loss gradients and small curvature are sufficient conditions for adversarial robustness. Second, we empirically show that input gradient regularization is competitive with adversarial training, even on non-smooth networks, at a fraction of the training time. Finally, we scale input gradient regularization to ImageNet-1k by using finite differences to estimate the gradient regularization term, rather than double backpropagation. This allows us to train adversarially robust networks on ImageNet-1k in 33 hours on four consumer grade GPUs.

5.2 Adversarial robustness bounds from the loss

5.2.1 *Background*

Much effort has been directed towards determining theoretical lower bounds on the minimum sized perturbation necessary to perturb an image so that it is misclassified by a model. One promising approach, proposed by Hein and Andriushchenko [HA17] and Weng et al. [WZC+18], and which has scaled well to ImageNet-1k, is to use the Lipschitz constant of the model. In this section, we build upon these ideas: we propose using the Lipschitz constant of a suitable loss, designed to measure classification errors. In addition, when the loss is twice continuously differentiable, we propose a second-order bound based on the maximum curvature of the loss.

Our notation is as follows. Write $y = f(x; w)$ for a model which takes input vectors x to label probabilities, with parameters w. Let $\mathcal{L}(y_1, y_2)$ be the loss and write $\ell(x) := \mathcal{L}(f(x, w), y)$, for the loss of a model f.

Finding an adversarial perturbation is interpreted as a global minimization problem: find the closest image to a clean image, in some specified norm, that is is also misclassified by the model

$$\min_v \|v\| \quad \text{subject to } f(x+v) \text{ misclassified} \tag{5.2}$$

However, (5.2) is a difficult and costly non-smooth, non-convex optimization problem. Instead, Goodfellow et al. [GSS14] proposed solving a surrogate problem: find a per-

turbation v of a clean image x that maximizes the loss, subject to the condition that the perturbation be inside a norm-ball of radius δ around the clean image. The surrogate problem is written

$$\max_v \ell(x+v) - c(v); \quad \text{where } c(v) = \begin{cases} 0 & \text{if } \|v\| \leq \delta \\ \infty & \text{otherwise} \end{cases} \tag{5.3}$$

The hard constraint $c(v)$ forces perturbations to be inside the norm-ball centred at the clean image x. Ideally, solutions of this surrogate problem (5.3) will closely align with solutions of the original more difficult global minimization problem. However, the hard constraint in (5.3) forces a particular scale: it may miss attacks which would succeed with only a slightly bigger norm. Additionally, the maximization problem (5.3) does not force misclassification, it only asks that the loss be increased.

The advantage of (5.3) is that it may be solved with gradient-based methods: present best-practice is to use variants of projected gradient descent (PGD), such as the iterative fast-signed gradient method [KGB16, MMS+17] when attacks are measured in the ℓ_∞ norm. However, gradient-based methods are not always effective: on non-smooth networks, such as those built of ReLU activation functions, a small gradient does not guarantee that the loss remains small locally. This deficiency was identified in [PMJ+16]. See Figure 5.1: ReLU networks may increase rapidly with a very small perturbation, even when local gradients are small. PGD methods will fail to locate these worst-case perturbations, and give a false impression of robustness. Carlini and Wagner [CW17b] avoid this scenario by incorporating decision boundary information into the loss; others solve (5.2) directly [BRB18, CJ19, FPO19].

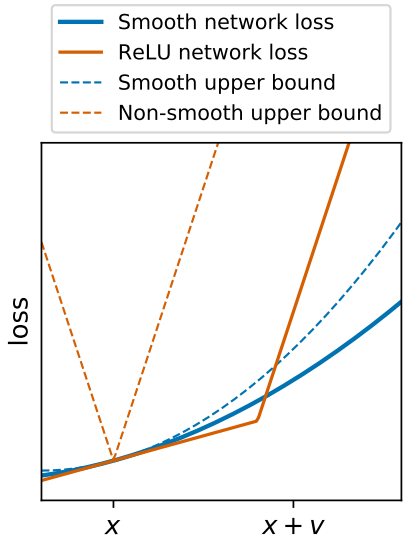


Figure 5.1: Illustration of upper bounds on the loss of two networks. For smooth networks (blue) with finite curvature, the loss is bounded above using $\ell(x)$ and $\nabla_x \ell(x)$. Non-smooth networks (orange) may have jumps in their gradients, which means robustness is not guaranteed by small local gradients.

5.2.2 *Derivation of lower bounds*

This leads us to consider the following compromise between (5.2) and (5.3). Consider the following modification of the Carlini-Wagner loss [CW17b] $\ell(x) = \max_{i \neq c} f_i(x) - f_c(x)$,

where c is the index of the correct label, and $f_i(x)$ is the model output for the i-th label. This loss has the appealing property the sign of the loss determines if the classification is correct. Adversarial attacks are found by minimizing

$$\min_v \|v\| \quad \text{subject to } \ell(x+v) \geq \ell_0 \tag{5.4}$$

The constant ℓ_0 determines when classification is incorrect; for the modified Carlini-Wagner loss, $\ell_0 = 0$. Problem (5.4) is closer to the true problem (5.2), and will always find an adversarial image. We use (5.4) to derive theoretical lower bounds on the minimum size perturbation necessary to misclassify an image. Suppose the loss is L-Lipschitz with respect to model input. Then we have the estimate

$$\ell(x+v) \leq \ell(x) + L\|v\| \tag{5.5}$$

Now suppose v *is* adversarial, with minimum adversarial loss $\ell(x+v) = \ell_0$. Then rearranging (5.5), we obtain the lower bound $\|v\| \geq \frac{1}{L}\left(\ell_0 - \ell(x)\right)$.

Unfortunately, the Lipschitz constant is a global quantity, and ignores local gradient information; see for example Huster et al. [HCC18]. Thus this bound can be quite poor, even when networks have small Lipschitz constant. On the other hand, if the model is twice continuously differentiable, then the loss landscape is smoother. This allows us to achieve a tighter bound, using local gradient information, as illustrated in Figure 5.1. Let C be an upper bound on the maximum positive eigenvalue of the Hessian of the loss over all x

$$C := \left(\max_x \lambda_{\max}(\nabla_x^2 \ell(x))\right)^+ \tag{5.6}$$

This value will be estimated empirically by maximizing over the dataset. The constant C is a measure of the largest positive curvature of the network. Using a Taylor approximation about x, we may upper bound the perturbed loss with

$$\ell(x+v) \leq \ell(x) + \langle v, \nabla_x \ell(x)\rangle + \frac{C}{2}\|v\|_2^2 \tag{5.7}$$

These two bounds give us the following.

Proposition 5.2.1. *Suppose the loss $\ell(x)$ is Lipschitz continuous with respect to model input x, with Lipschitz constant L. Let ℓ_0 be such that if $\ell(x) < \ell_0$, the model is always correct. Then a lower bound on the minimum magnitude of perturbation v necessary to adversarially perturb an image x is*

$$\|v\| \geq \frac{\max\{\ell_0 - \ell(x), 0\}}{L} \tag{L-bound}$$

Suppose in addition that the loss is twice-differentiable, with maximum curvature C (defined as in

(5.6)). *Then*

$$\|v\|_2 \geq \frac{1}{C}\left(-\|\nabla\ell(x)\|_2 + \sqrt{\|\nabla\ell(x)\|_2^2 + 2C\max\{\ell_0 - \ell(x), 0\}}\right) \qquad (C\text{-bound})$$

The proof of (L-bound) is given above; the proof of (C-bound) follows by rearranging (5.7) and solving for $\|v\|$.

Remark 5.1. The second-order bound requires that the network and loss are smooth with respect to the input, but almost all image classification networks now use ReLUs, which are not smooth. We use the following smoothed ReLU

$$\sigma(x) = \begin{cases} \max(x, 0) & \text{if } |x| \geq \frac{1}{2} \\ -\frac{1}{2}\left(x + \frac{1}{2}\right)^4 + \left(x + \frac{1}{2}\right)^3 & \text{if } |x| < \frac{1}{2} \end{cases} \qquad (5.8)$$

This activation function is twice continuously differentiable, and avoids the vanishing gradient problem of smooth sigmoidal activation functions. Moreover because it agrees with $\mathrm{ReLU}(x)$ outside of the interval $(-\frac{1}{2}, \frac{1}{2})$, it is fairly efficient during backpropagation. As for the loss, a smooth version of the Carlini-Wagner loss is available by using a soft maximum, rather than a strict $\max$.

Proposition 5.2.1 motivates the need for input gradient regularization. The Lipschitz constant L is the maximum gradient norm of *the loss* over all inputs. Therefore (L-bound) says that a regularization term encouraging small gradients (and so reducing L) should increase the minimum adversarial distance. This aligns with [HA17], who proposed the cross-Lipschitz regularizer, penalizing networks with large Jacobians in order to shrink the Lipschitz constant of *the network*.

However, this is not enough: the gap $\ell_0 - \ell(x)$ must be large as well. This explains one form of 'gradient masking' [PMG+17]. Shrinking the magnitude of gradients while also closing the gap $\ell_0 - \ell(x)$ effectively does nothing to improve adversarial robustness. For example, in defense distillation, the magnitude of the model Jacobian is reduced by increasing the temperature of the final softmax layer of the network. However, this has the detrimental side-effect of sending the model output to $(\frac{1}{N}, \cdots, \frac{1}{N})$, where N is the number of classes, which effectively shrinks the loss gap to zero. Thus with high distillation temperatures the lower bound provided by Proposition 5.2.1 approaches zero.

Moreover, even supposing the loss gradients are small and the gap $\ell_0 - \ell(x)$ is large, there may still be adversarially vulnerable images. For example, suppose we have two smooth networks, one with large curvature, and another with small curvature. Suppose that there is an image with zero gradient on both networks, each with identically large loss gaps $\ell_0 - \ell(x)$. The second-order bound (C-bound) says that the minimum adversarial

distance here is bounded below by $\|v\| \geq \sqrt{\max\{\ell_0 - \ell(x), 0\}/C}$. In other words, the network with smaller curvature is more robust.

Taken together, Proposition 5.2.1 provides three sufficient conditions for training robust networks: (i) the loss gap $\ell_0 - \ell(x)$ should be large; (ii) the gradients of the loss should be small; and (iii) the curvature of the loss should also be small. The first point will be satisfied by default when the loss is minimized. The second point will be satisfied by training with a loss regularized to penalize large input gradients. Experimentally the third point is satisfied with input gradient regularization. When these conditions are satisfied, local information is enough to guarantee robustness.

Our robustness bounds are most similar in spirit to Weng et al. [WZC+18], who derive bounds using an estimate of the *local* Lipschitz constant of the model. Moosavi-Dezfooli et al. [MFUF18] have also used a second order approximation to derive approximate robustness bounds for binary classification, but they neglected higher order error terms. Cohen et al. [CRK19] derive bounds by training with normally distributed input noise, then averaging model predictions normally sampled about the input image. It is well known that training with normal noise is equivalent to squared ℓ_2 norm gradient regularization [Bis95]; thus Cohen et al. [CRK19] achieve gradient regularization indirectly. Our bounds require at most one gradient and model evaluation per image once L and C have been estimated; whereas both Cohen et al. and Weng et al. require many hundreds of local model evaluations per image. Since L and C are globally estimated, our bounds could be improved using these local sampling techniques to obtain *local* values of L and C, with more computational effort.

5.3 Squared norm gradient regularization

Proposition 5.2.1 provides strong motivation for input gradient regularization as a method for promoting adversarial robustness. However, it does not tell us what form the gradient regularization term should take. In this section, we show how norm squared gradient regularization arises from a *quadratic cost*.

In adversarial training, solutions of (5.3) are used to generate images on which the network is trained. In effect, adversarial training seeks a solution of the minimax problem

$$\min_w \mathop{\mathbb{E}}_{x\sim\mathbb{P}} \left[\max_v \ell(x+v; w) - c(v)\right] \tag{5.9}$$

where $\mathbb{P}$ is the distribution of images. This is a robust optimization problem [Wal45, RL87]. The cost function $c(v)$ penalizes perturbed images from being too far from the original. When the cost function is the hard constraint from (5.3), perturbations must be inside a

norm ball of radius δ. This leads to adversarial training with PGD [KGB16, MMS+17]. However this forces a particular scale: it is possible that no images are adversarial within radius δ, but that there are adversarial images with only a slightly larger distance. Instead of using a hard constraint, we can relax the cost function to be the quadratic cost $c(v) = \frac{1}{2\delta}\|v\|^2$. The quadratic cost allows attacks to be of any size, but penalizes larger attacks more than smaller attacks. With a quadratic cost, there is less of a danger that a local attack will be overlooked.

Solving (5.9) directly is expensive: on ImageNet-1k, both Kannan et al. [KKG18] and Xie et al. [XWvdM+18] required large-scale distributed training with many dozens or hundreds of GPUs, and over a week of training time. Instead we take the view that (5.9) may be bounded above, and solved approximately. When the loss is smooth and $c(v) = \frac{1}{2\delta}\|v\|^2$, the optimal value of $\max_v \ell(x+v) - c(v)$ using the bound (5.7) is $\frac{\delta}{2(1-\delta C)}\|\nabla_x \ell(x)\|_*^2$, provided $\delta < \frac{1}{C}$. This gives the following proposition.

Proposition 5.3.1. *Suppose both the model and the loss are twice continuously differentiable. Suppose attacks are measured with quadratic cost $\frac{1}{2\delta}\|v\|^2$. Then the optimal value of (5.9) is bounded above by*

$$\min_w \mathop{\mathbb{E}}_{x\sim\mathbb{P}} \left[\ell(x; w) + \frac{\lambda}{2}\|\nabla_x \ell(x)\|_*^2 \right] \tag{5.10}$$

where $\lambda = \frac{\delta}{1-\delta C}$.

That is, we may bound the solution of the adversarial training problem (5.9) by solving the gradient regularization problem (5.10), when the cost function is quadratic. It is not necessary to know δ or compute C; they are absorbed into λ. In the adversarial robustness literature, input gradient regularization using the squared ℓ_2 norm was proposed by Ross and Doshi-Velez [RD18]. It was expanded by Roth et al. [RLNH18] to use a Mahalanobis norm with the correlation matrix of adversarial attacks. When $c(v)$ is the hard constraint forcing attacks inside the δ norm ball and C is small, supposing the curvature term is negligible, we can estimate the maximum in (5.9) by $\ell(x) + \frac{1}{\delta}\|\nabla_x \ell(x)\|_*$, using the dual norm for the gradient. This is norm gradient regularization (not squared), and was recently used for adversarial robustness on both CIFAR-10 [SOS+18], and MNIST [SLC19].

5.3.1 *Finite difference implementation*

Norm squared input gradient regularization has long been used as a regularizer in neural networks: Drucker and LeCun [DL91] first showed its effectiveness for generalization. Drucker and LeCun [DL91] implemented gradient regularization with 'double backpropa-

gation' to compute the derivatives of the penalty term with respect to the model parameters w, which is needed to update the parameters during training. Double backpropagation involves two passes of automatic differentiation: one pass to compute the gradient of the loss with respect to the inputs x, and another pass on the output of the first to compute the gradient of the penalty term with respect to model parameters w. In neural networks, double backpropagation is the standard technique for computing the parameter gradient of a regularized loss. However, it is not currently scaleable to large neural networks. Instead we approximate the gradient regularization term with finite differences.

Proposition 5.3.2 (Finite difference approximation of squared ℓ_2 gradient norm)**.** *Let d be the normalized input gradient direction: $d = \nabla_x \ell(x)/\|\nabla_x \ell(x)\|_2$ when the gradient is nonzero, and set $d = 0$ otherwise. Let h be the finite difference step size. Assume further that the loss is twice continuously differentiable. Then, the squared ℓ_2 gradient norm is approximated by*

$$\|\nabla_x \ell(x)\|_2^2 \approx \left(\frac{\ell(x + hd) - \ell(x)}{h}\right)^2 \tag{5.11}$$

The vector d is normalized to ensure the accuracy of the finite difference approximation, which is of order h, as can be seen by a Taylor approximation. The finite differences approximation (5.11) allows the computation of the gradient of the regularizer (with respect to model parameters w) to be done with only two regular passes of backpropagation, rather than with double backpropagation. On the first, the input gradient direction d is calculated. The second computes the gradient with respect to model parameters by performing backpropagation on the right-hand-side of (5.11). Double backpropagation is avoided by detaching d from the computational graph after the first pass. In practice, for large networks, we have found that the finite difference approximation of the regularization term is considerably more efficient than using double backpropagation.

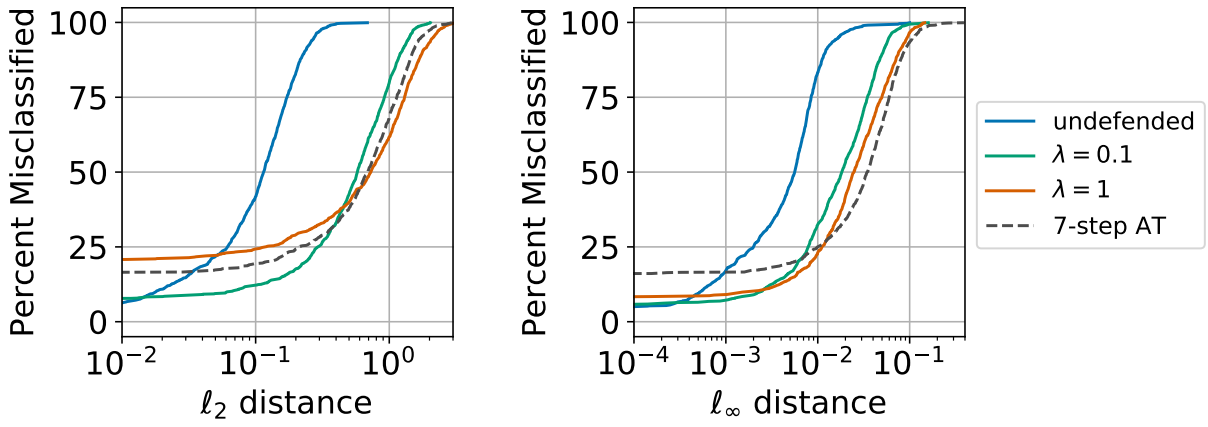
The proposed training algorithm, with squared Euclidean input gradient regularization, is presented in Algorithm 2. Other gradient penalty terms can be approximated as well. For example, when defending against attacks measured in the ℓ_∞ norm, the squared ℓ_1 norm penalty can approximated by setting instead $d = \operatorname{sgn}(\nabla_x \ell(x))/\sqrt{N}$ when the gradient is nonzero.

5.4 Experimental results

In this section we provide empirical evidence that input gradient regularization is an effective tool for promoting adversarial robustness, *even on non-smooth networks* built with standard ReLU activation functions.

Algorithm 2 Training with squared ℓ_2-norm input gradient regularization, using finite differences

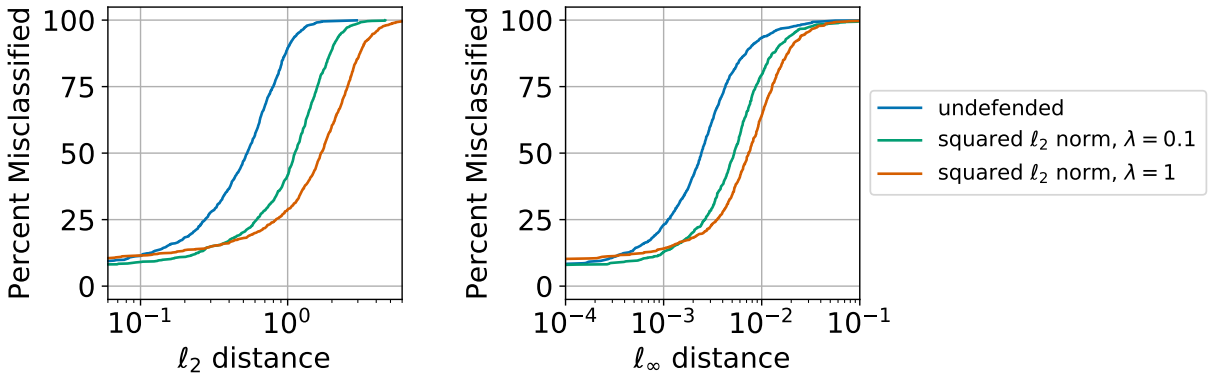
Input: Initial model parameters w_0
Hyperparameters: Regularization strength λ; batch size m; finite difference discretization h
while w_t not converged **do**
  sample minibatch of data $\left\{(x^{(i)}, y^{(i)})\right\}_{i=1,\dots,m}$ from empirical distribution $\hat{\mathbb{P}}$
  for $i = 0$ **to** m **do**
    $g^{(i)} = \nabla_x \ell(x^{(i)}, y^{(i)}; w_t)$
    $d^{(i)} = \begin{cases} \frac{g^{(i)}}{\|g^{(i)}\|_2} & \text{if } g^{(i)} \neq 0 \\ 0 & \text{otherwise} \end{cases}$ ▷ for ℓ_1-norm use normalized signed gradient
    detach $d^{(i)}$ from computational graph
    $z^{(i)} = x^{(i)} + h d^{(i)}$
  end for
  $\mathcal{L}(w) = \frac{1}{m} \sum_{i=1}^{m} \ell(x^{(i)}, y^{(i)}; w)$
  $\mathcal{R}(w) = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{h^2} \left(\ell(z^{(i)}, y^{(i)}; w) - \ell(x^{(i)}, y^{(i)}; w)\right)^2$
  $w_{t+1} \leftarrow w_t - \tau_t \nabla_w \left(\mathcal{L}(w_t) + \lambda \mathcal{R}(w_t)\right)$
end while



a ℓ_2 *norm adversarial attacks*

b ℓ_∞ *norm adversarial attacks*

Figure 5.2: Adversarial attacks on the CIFAR-10 dataset, on networks built with standard ReLUs. Regularized networks attacked in ℓ_2 are trained with squared ℓ_2 norm gradient regularization; networks attacked in ℓ_∞ are trained with squared ℓ_1 norm regularization.



a *ℓ_2-norm adversarial attacks* **b** *ℓ_∞-norm adversarial attacks*

Figure 5.3: Adversarial attacks on ImageNet-1k with the ResNet-50 architecture. Top5 error reported.

We train networks on the CIFAR-10 dataset [KH09], and ImageNet-1k [DDS+09]. On the CIFAR dataset we use the ResNeXt architecture[1] [XGD+17]; on ImageNet-1k we use a ResNet-50 [HZRS16]. The CIFAR networks were trained with standard data augmentation and learning rate schedules on a single GeForce GTI 1080 Ti. On ImageNet-1k, we modified the training code of Shaw et al.'s [SBH] submission to the DAWNBench competition [CKN+18] and train with four GPUs. Training code and trained model weights will be made available.

We train an undefended network as a baseline to compare various types of regularization. On CIFAR-10, networks are trained with squared ℓ_2 and squared ℓ_1 gradient norm regularization. The former is appropriate for defending against attacks measured in ℓ_2; the latter for attacks measured in ℓ_∞. We set the regularization strength to be either $\lambda = 0.1$ or 1; and set finite difference discretization $h = 0.01$. We compare each network with the current state-of-the-art form of adversarial training, with models trained using the hyperparameters in Madry et al. [MMS+17] (7-steps of FGSM, ℓ_∞ step size $\frac{2}{255}$, projected onto an ℓ_∞ ball of radius $\frac{8}{255}$). On ImageNet-1k we only train adversarially robust models with squared ℓ_2 regularization.

On each dataset, we attack 1000 randomly selected images. We perturb each image with attacks in both the Euclidean and ℓ_∞ norms, with a suite of current state-of-the-art attacks: the Carlini-Wagner attack [CW17b]; the Boundary attack [BRB18]; the LogBarrier attack [FPO19]; and PGD [MMS+17] (in both the ℓ_∞ norm or the ℓ_2 norm). The former three attacks are effective at evading gradient masking defences; the latter is very good at finding images close to the original when gradients are not close to zero. We record the

[1] ResNeXt34-2x32 on CIFAR-10; ResNeXt34-2x64 on CIFAR-100

Table 5.1: Adversarial robustness statistics, measured in the ℓ_∞ norm. Top1 error is reported on CIFAR-10; Top5 error on ImageNet-1k.

	smooth ReLU?	% clean error	% error at $\varepsilon = \frac{2}{255}$	% error at $\varepsilon = \frac{8}{255}$	mean distance	improvement ratio	training time (hours)
CIFAR-10							
Undefended		**4.36**	70.82	98.94	6.62e−3	-	2.06
Madry et al (7-step AT)		16.33	22.86	**46.02**[2]	4.07e−2	1.88	12.10
squared ℓ_1 norm, $\lambda = 0.1$		6.45	24.92	70.41	2.35e−2	**5.31**	5.22
squared ℓ_1 norm, $\lambda = 1$		9.02	**18.47**	58.69	3.34e−2	3.78	5.15
ImageNet-1k							
Undefended		**6.94**	90.21	98.94	3.94e−3	-	20.30
Undefended	✓	9.39	82.03	95.42	9.74e−3	4.17	23.46
squared ℓ_2 norm, $\lambda = 0.1$		7.66	70.56	97.53	7.96e−3	**9.83**	32.60
squared ℓ_2 norm, $\lambda = 0.1$	✓	9.49	63.23	**94.21**	1.24e−2	5.84	52.47
squared ℓ_2 norm, $\lambda = 1$		10.26	**52.79**	95.93	9.95e−3	3.19	33.87

best adversarial distance *on a per image basis*, for each norm.

Adversarial robustness results for networks attacked in the ℓ_∞ norm are presented in Table 5.1. These results are for networks built of standard ReLUs. Table 5.1 and Figures 5.2 and 5.3 demonstrate a clear trade-off between test accuracy and adversarial robustness, as the strength of the regularization is increased. On CIFAR-10, the undefended network achieves test error of 4.36%, but is not robust to attacks even at ℓ_∞ distance $\frac{2}{255}$. However with a strong regularization parameter ($\lambda = 1$), test error increases to 9.02% on clean images, and only 18.47% test error at attack distance $\frac{2}{255}$. In contrast, the network trained with 7-steps of adversarial training appears to be over-regularized: on clean images, the adversarially trained network achieves 16.33% test error, but 22.86% error at distance $\frac{2}{255}$. To be fair, at the commonly reported ℓ_∞ of $\frac{8}{255}$, the adversarially trained network outperforms the best gradient regularized networks by about 12%, but at over twice the training time of the regularized networks. On ImageNet, we see a reduction of nearly 40% at distance $\frac{2}{255}$.

It has been noted that adversarial robustness comes with a cost of degraded test error [TSE+18]. This trade-off may be quantified. We measure the relative improvement in adversarial robustness against the cost of degraded test error with the following metric. Suppose an undefended network has test error e_0, and let a regularized network's network

[2]Madry et al report 54.2% error at $\varepsilon = \frac{8}{255}$ with the WRN-28x10 architecture; our results are obtained with ResNeXt34 (2x32).

test error be denoted e_λ. Define the relative degradation in test error to be $R_e = (e_\lambda - e_0)/e_0$. Similarly define the relative improvement in robustness (measured by mean adversarial distance μ) to be $R_\mu = (\mu_\lambda - \mu_0)/\mu_0$. We define the *adversarial improvement ratio* to be R_μ/R_e. This measures the improvement in adversarial robustness against the expense of poorer test error: high values mean the defended model is much more robust and has not lost significant test accuracy. Values close to zero imply the model is more robust but has a much worse test accuracy relative to the undefended model. The improvement ratio is non-dimensional, and so it allows for comparison between datasets.

Measured in this metric, the tradeoff between test accuracy and adversarial robustness is clear. On both ImageNet-1k and CIFAR-10, models regularized with $\lambda = 0.1$ offer the best trade-off between robustness and test error. If test accuracy is not of foremost concern, then stronger regularization parameters may be chosen. If neither training time nor test accuracy are important factors, then adversarial training is competitive with gradient regularization.

In Table 5.3 we report results on models trained for attacks in the ℓ_2 norm. On CIFAR-10, the most robust model is trained with regularization strength $\lambda = 1$, and outperforms even the adversarially trained model. On ImageNet-1k, we see the same pattern: the model trained with $\lambda = 1$ offers the best protection against adversarial attacks. Due to the long training time, we were not able to train ImageNet-1k with multi-step adversarial training.

In Table 5.3 we also report our theoretical bounds on the minimum distance required to adversarially perturb, using the Carlini-Wagner loss.[3] Figures 5.4 and 5.5 show these bounds on a per-image basis. The theoretical bounds require calculating constants L and C, which are not readily available. Instead, we estimate L as the maximum gradient norm over test images; for smooth models we estimate C as the maximum spectral norm of the Hessian.[4] These estimates are reported in Table 5.2. Gradient regularization reduces L and C, by *one to two orders of magnitude*. Table 5.2 shows adversarial training also reduces L: effectively adversarial training is a regularizer. Because L and C are estimated, and not exact, one would expect that our bounds would sometimes fail. However, on CIFAR-10, the bounds reliable held on all attacked images. On ImageNet-1k, the bounds failed on about 9% of attacked test images, which indicates that C and L could be estimated more accurately, for example using by estimating these constants locally like in [WZC+18].

[3] This loss can be modified for Top-5 mis-classification as well.

[4] We compute the spectral norm of the Hessian using the Lanczos algorithm [GVL12, §10.1] on Hessian-vector products (computed via automatic differentiation).

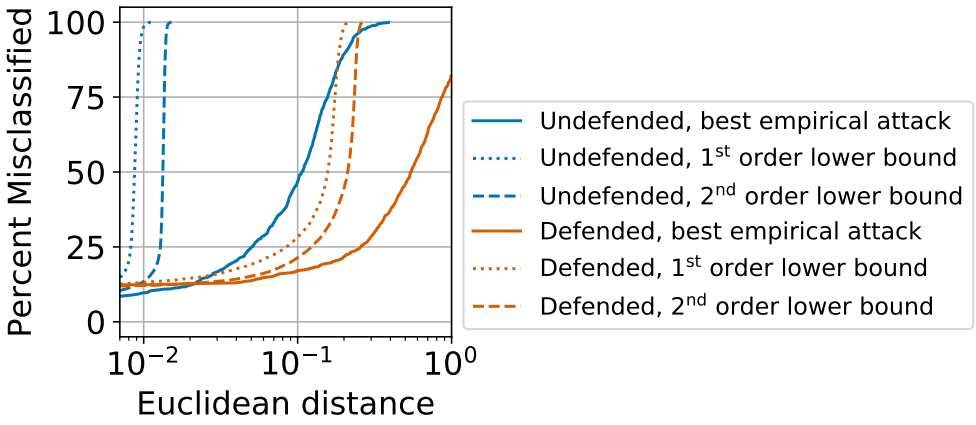


Figure 5.4: Theoretical minimum lower bound on adversarial distance for CIFAR-10, on networks with smooth ReLU activation functions. Defended networks trained with $\lambda = 0.1$, penalized with squared ℓ_2 norm gradient.

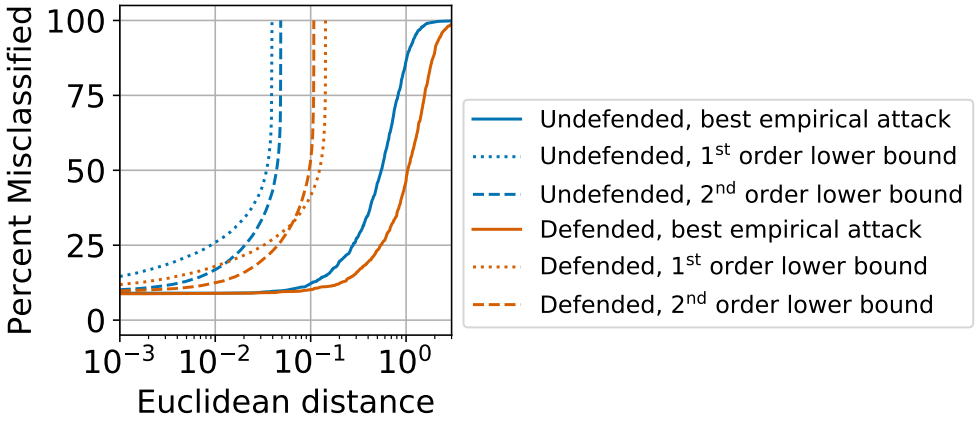


Figure 5.5: Theoretical minimum lower bound on adversarial distance for ImageNet-1k, on networks with smooth ReLU activation functions. Defended networks trained with $\lambda = 0.1$, penalized with squared ℓ_2 norm gradient.

5.5 CONCLUSION

We have provided motivation for training adversarially robust networks through input gradient regularization, by bounding the minimum adversarial distance with gradient and curvature statistics of the loss. We have shown empirically that gradient regularization is scaleable to ImageNet-1k, and provides adversarial robustness competitive with adversarial training. We gave theoretical per-image bounds on the minimum adversarial distance, for non-smooth models (using the Lipschitz constant of the loss), and augmented these bounds using smooth models with a second-order bound based on model curvature. These bounds were empirically validated against state-of-the-art attacks.

Table 5.2: Regularity statistics on selected models, measured in the ℓ_2 norm. Statistics computed using modified loss $\max_{i \neq c} f_i(x) - f_c(x)$. A soft maximum is used for curvature statistics.

	soft ReLU?	mean		maximum	
		$\|\nabla \ell(x)\|$	$\|\nabla^2 \ell(x)\|$	$\|\nabla \ell(x)\|$	$\|\nabla^2 \ell(x)\|$
CIFAR-10					
Undefended		3.05	-	122.34	-
Undefended	✓	3.25	198.23	65.35	8134.26
Madry et al (7-step AT)		0.40	-	2.52	-
squared ℓ_2 norm, $\lambda = 0.1$		0.58	-	4.43	-
squared ℓ_2 norm, $\lambda = 0.1$	✓	0.65	2.08	4.52	27.05
squared ℓ_2 norm, $\lambda = 1$		0.35	-	1.33	-
ImageNet-1k					
Undefended		1.12	-	17.51	-
Undefended	✓	1.02	11.61	25.43	848.69
squared ℓ_2 norm, $\lambda = 0.1$		0.46	-	4.85	-
squared ℓ_2 norm, $\lambda = 0.1$	✓	0.45	1.87	6.99	171.98
squared ℓ_2 norm, $\lambda = 1$		0.27	-	2.12	-

Table 5.3: Adversarial robustness statistics, measured in ℓ_2. Top1 error is reported on CIFAR-10; Top5 error on ImageNet-1k.

	smooth ReLU?	% clean error	mean adversarial distance			improvement ratio	training time (hours)
			L-bound	C-bound	empirical		
CIFAR-10							
Undefended		**4.36**	5.57e−3	-	0.12	-	2.06
Undefended	✓	6.84	1.01e−2	1.19e−2	0.11	−0.20	3.78
Madry et al (7-step AT)		16.33	0.18	-	0.74	1.81	12.10
squared ℓ_2 norm, $\lambda = 0.1$		8.03	0.14	-	0.63	**4.86**	5.18
squared ℓ_2 norm, $\lambda = 0.1$	✓	11.68	0.13	0.17	0.59	2.25	9.46
squared ℓ_2 norm, $\lambda = 1$		20.31	**0.30**	-	**0.81**	1.52	5.08
ImageNet-1k							
Undefended		**6.94**	3.63e−2	-	0.55	-	20.30
Undefended	✓	9.39	2.56e−2	3.40e−2	0.56	0.12	23.46
squared ℓ_2 norm, $\lambda = 0.1$		7.66	0.13	-	1.14	**10.23**	32.60
squared ℓ_2 norm, $\lambda = 0.1$	✓	9.49	9.23e−2	7.52e−2	1.09	2.64	52.47
squared ℓ_2 norm, $\lambda = 1$		10.26	**0.26**	-	**1.75**	4.52	33.87

BIBLIOGRAPHY

[ACW18] Anish Athalye, Nicholas Carlini, and David Wagner, *Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples*, Proceedings of the 35th International Conference on Machine Learning (Stockholmsmässan, Stockholm Sweden) (Jennifer Dy and Andreas Krause, eds.), Proceedings of Machine Learning Research, vol. 80, PMLR, 10–15 Jul 2018, pp. 274–283.

[AKM19] Scott Armstrong, Tuomo Kuusi, and Jean-Christophe Mourrat, *Quantitative stochastic homogenization and large-scale regularity*, Springer, 2019.

[AS14] Scott N. Armstrong and Charles K. Smart, *Quantitative Stochastic Homogenization of Elliptic Equations in Nondivergence Form*, Archive for Rational Mechanics and Analysis **214** (2014), no. 3, 867–911 (en).

[BCM+13] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli, *Evasion attacks against machine learning at test time*, Machine Learning and Knowledge Discovery in Databases (Berlin, Heidelberg) (Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip Železný, eds.), Springer Berlin Heidelberg, 2013, pp. 387–402.

[BCM16] Jean-David Benamou, Francis Collino, and Jean-Marie Mirebeau, *Monotone and consistent discretization of the monge-ampere operator*, Mathematics of computation **85** (2016), no. 302, 2743–2775.

[BEJ84] Emmanuel Nicholas Barron, Lawrence Craig Evans, and Robert Jensen, *Viscosity solutions of isaacs' equations and differential games with lipschitz controls*, Journal of Differential Equations **53** (1984), no. 2, 213–233.

[Bis95] Christopher M. Bishop, *Training with noise is equivalent to Tikhonov regularization*, Neural Computation **7** (1995), no. 1, 108–116.

[BLP11] Alain Bensoussan, Jacques-Louis Lions, and George Papanicolaou, *Asymptotic analysis for periodic structures*, vol. 374, American Mathematical Soc., 2011.

[BPR16] Olivier Bokanowski, Athena Picarelli, and Christoph Reisinger, *High-order filtered schemes for time-dependent second order hjb equations*, arXiv preprint arXiv:1611.04939 (2016).

[BRB18] Wieland Brendel, Jonas Rauber, and Matthias Bethge, *Decision-based adversarial attacks: Reliable attacks against black-box machine learning models*, 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, 2018.

[BS91] Guy Barles and Panagiotis E. Souganidis, *Convergence of approximation schemes for fully nonlinear second order equations*, Asymptotic Anal. **4** (1991), no. 3, 271–283.

[Cal17] Jeff Calder, *Consistency of Lipschitz learning with infinite unlabeled data and finite labeled data* (en).

[Cal19] Jeffrey W Calder, *The game theoretic p-laplacian and semi-supervised learning with few labels*, Nonlinearity **32** (2019), no. 1, 301–330 (English (US)).

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

[CC95] Luis A Caffarelli and Xavier Cabré, *Fully nonlinear elliptic equations*, vol. 43, American Mathematical Soc., 1995.

[CC16] Simone Cacace and Fabio Camilli, *Ergodic problems for Hamilton-Jacobi equations: yet another but efficient numerical method*, arXiv preprint arXiv:1601.07107 (2016).

[CEL84] Michael G Crandall, Lawrence C Evans, and P-L Lions, *Some properties of viscosity solutions of hamilton-jacobi equations*, Transactions of the American Mathematical Society **282** (1984), no. 2, 487–502.

[CG08] LA Caffarelli and Roland Glowinski, *Numerical solution of the Dirichlet problem for a Pucci equation in dimension two. Application to homogenization*, Journal of Numerical Mathematics **16** (2008), no. 3, 185–216.

[CIL92] Michael G. Crandall, Hitoshi Ishii, and Pierre-Louis Lions, *User's guide to viscosity solutions of second order partial differential equations*, Bull. Amer. Math. Soc. (N.S.) **27** (1992), no. 1, 1–67.

[CJ19] Jianbo Chen and Michael I. Jordan, *Boundary attack++: Query-efficient decision-based adversarial attack*, CoRR **abs/1904.02144** (2019).

[CKN+18] Cody Coleman, Daniel Kang, Deepak Narayanan, Luigi Nardi, Tian Zhao, Jian Zhang, Peter Bailis, Kunle Olukotun, Christopher Ré, and Matei Zaharia, *Analysis of dawnbench, a time-to-accuracy machine learning performance benchmark*, CoRR **abs/1806.01427** (2018).

[CL83] Michael G Crandall and Pierre-Louis Lions, *Viscosity solutions of hamilton-jacobi equations*, Transactions of the American Mathematical Society **277** (1983), no. 1, 1–42.

[CM09] Fabio Camilli and Claudio Marchi, *Rates of convergence in periodic homogenization of fully nonlinear uniformly elliptic PDEs*, Nonlinearity **22** (2009), no. 6, 1481.

[CRK19] Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter, *Certified adversarial robustness via randomized smoothing*, CoRR **abs/1902.02918** (2019).

[CS10] Luis A Caffarelli and Panagiotis E Souganidis, *Rates of convergence for the homogenization of fully nonlinear uniformly elliptic pde in random media*, Inventiones mathematicae **180** (2010), no. 2, 301–360.

[CSW05] Luis A Caffarelli, Panagiotis E Souganidis, and Lihe Wang, *Homogenization of fully nonlinear, uniformly elliptic and parabolic partial differential equations in stationary ergodic media*, Communications on pure and applied mathematics **58** (2005), no. 3, 319–361.

[CW17a] Nicholas Carlini and David A. Wagner, *Adversarial examples are not easily detected: Bypassing ten detection methods*, Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, AISec@CCS 2017, Dallas, TX, USA, November 3, 2017, 2017, pp. 3–14.

[CW17b] ———, *Towards evaluating the robustness of neural networks*, 2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017, 2017, pp. 39–57.

[CW17c] Yangang Chen and Justin WL Wan, *Multigrid methods for convergent mixed finite difference scheme for monge–ampère equation*, Computing and Visualization in Science (2017), 1–15.

[CWL16] Yangang Chen, Justin WL Wan, and Jessey Lin, *Monotone mixed finite difference scheme for monge–ampère equation*, Journal of Scientific Computing (2016), 1–29.

[DB08] Germund Dahlquist and Åke Björck, *Numerical methods in scientific computing, volume i*, Society for Industrial and Applied Mathematics **8** (2008).

[DDS+09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li, *Imagenet: A large-scale hierarchical image database*, 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA, 2009, pp. 248–255.

[DG05] Edward J Dean and Roland Glowinski, *On the numerical solution of a two-dimensional pucci's equation with dirichlet boundary conditions: a least-squares approach*, Comptes Rendus Mathematique **341** (2005), no. 6, 375–380.

[DL91] Harris Drucker and Yann LeCun, *Double backpropagation increasing generalization performance*, IJCNN-91-Seattle International Joint Conference on Neural Networks, vol. 2, IEEE, 1991, pp. 145–150.

[ES08] Björn Engquist and Panagiotis E Souganidis, *Asymptotic and numerical homogenization*, Acta Numerica **17** (2008), 147–190.

[Eva80] Lawrence C. Evans, *On solving certain nonlinear partial differential equations by accretive operator methods*, Israel Journal of Mathematics **36** (1980), no. 3, 225–247.

[Eva82] Lawrence C Evans, *Classical solutions of fully nonlinear, convex, second-order elliptic equations*, Communications on Pure and Applied Mathematics **35** (1982), no. 3, 333–363.

[Eva89] ———, *The perturbed test function method for viscosity solutions of nonlinear PDE*, Proceedings of the Royal Society of Edinburgh: Section A Mathematics **111** (1989), no. 3-4, 359–375.

[Eva92] ———, *Periodic homogenisation of certain fully nonlinear partial differential equations*, Proceedings of the Royal Society of Edinburgh: Section A Mathematics **120** (1992), no. 3-4, 245–265.

[FFL+17] Zexin Feng, Brittany D. Froese, Rongguang Liang, Dewen Cheng, and Yongtian Wang, *Simplified freeform optics design for complicated laser beam shaping*, Appl. Opt. **56** (2017), no. 33, 9308–9314.

[FM14] Jérôme Fehrenbach and Jean-Marie Mirebeau, *Sparse Non-negative Stencils for Anisotropic Diffusion*, Journal of Mathematical Imaging and Vision **49** (2014), no. 1, 123–147 (English).

[FO09] Brittany D. Froese and Adam M. Oberman, *Numerical averaging of non-divergence structure elliptic operators*, Communications in Mathematical Sciences **7** (2009), no. 4, 785–804.

[FO13] ———, *Convergent filtered schemes for the Monge-Ampère partial differential equation*, SIAM J. Numer. Anal. **51** (2013), no. 1, 423–444. MR 3033017

[FO18a] Chris Finlay and Adam M. Oberman, *Approximate homogenization of convex nonlinear elliptic PDEs*, Communications in Mathematical Sciences **16** (2018), no. 7, 1895 – 1906.

[FO18b] ———, *Approximate Homogenization of Fully Nonlinear Elliptic PDEs: Estimates and Numerical Results for Pucci Type Equations*, Journal of Scientific Computing **77** (2018), no. 2, 936–949.

[FO18c] ———, *Improved accuracy of monotone finite difference schemes on point clouds and regular grids*, arXiv:1807.05150 [math] (2018), arXiv: 1807.05150.

[FO19] ———, *Scaleable input gradient regularization for adversarial robustness*, arXiv:1905.11468 [cs, stat] (2019), arXiv: 1905.11468.

[FP07] Francisco Facchinei and Jong-Shi Pang, *Finite-dimensional variational inequalities and complementarity problems*, Springer Science & Business Media, 2007.

[FPO19] Chris Finlay, Aram-Alexandre Pooladian, and Adam M. Oberman, *The Log-Barrier adversarial attack: making effective use of decision boundary information*, CoRR **abs/1903.10396** (2019).

[Fro18] Brittany D. Froese, *Meshfree finite difference approximations for functions of the eigenvalues of the Hessian*, Numerische Mathematik **138** (2018), no. 1, 75–99.

[FS06] Wendell H Fleming and Halil Mete Soner, *Controlled markov processes and viscosity solutions*, vol. 25, Springer Science & Business Media, 2006.

[FS17] Brittany D Froese and Tiago Salvador, *Higher-order Adaptive Finite Difference Methods for Fully Nonlinear Elliptic Equations*, arXiv preprint arXiv:1706.07741 (2017).

[GB08] Michael Grant and Stephen Boyd, *Graph implementations for nonsmooth convex programs*, Recent Advances in Learning and Control (V. Blondel, S. Boyd, and H. Kimura, eds.), Lecture Notes in Control and Information Sciences, Springer-Verlag Limited, 2008, http://stanford.edu/~boyd/graph_dcp.html, pp. 95–110.

[GB14] ———, *CVX: Matlab software for disciplined convex programming, version 2.1*, http://cvxr.com/cvx, March 2014.

[GMP18] Ian Goodfellow, Patrick McDaniel, and Nicolas Papernot, *Making machine learning robust against adversarial inputs*, Communications of the ACM **61** (2018), no. 7, 56–66 (en).

[GO04] Diogo A Gomes and Adam M Oberman, *Computing the effective Hamiltonian using a variational approach*, SIAM journal on control and optimization **43** (2004), no. 3, 792–812.

[Gom05] Diogo Aguiar Gomes, *Trends in partial differential equations of mathematical physics*, ch. Duality Principles for Fully Nonlinear Elliptic Equations, pp. 125–136, Birkhäuser Basel, Basel, 2005.

[GSS14] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy, *Explaining and harnessing adversarial examples*, CoRR **abs/1412.6572** (2014).

[GVL12] Gene H Golub and Charles F Van Loan, *Matrix computations*, vol. 3, JHU press, 2012.

[HA17] Matthias Hein and Maksym Andriushchenko, *Formal Guarantees on the Robustness of a Classifier against Adversarial Manipulation*, Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, 2017, pp. 2263–2273.

[HCC18] Todd Huster, Cho-Yu Jason Chiang, and Ritu Chadha, *Limitations of the lipschitz constant as a defense against adversarial examples*, ECML PKDD 2018 Workshops - Nemesis 2018, UrbReas 2018, SoGood 2018, IWAISe 2018, and Green Data Mining 2018, Dublin, Ireland, September 10-14, 2018, Proceedings, 2018, pp. 16–29.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, *Identity mappings in deep residual networks*, Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV, 2016, pp. 630–645.

[IMT16] Hitoshi Ishii, Hiroyoshi Mitake, and Hung V. Tran, *The vanishing discount problem and viscosity mather measures. part 1: The problem on a torus*, Journal de Mathématiques Pures et Appliquées (2016).

[Jen88] Robert Jensen, *The maximum principle for viscosity solutions of fully nonlinear second order partial differential equations*, Archive for Rational Mechanics and Analysis **101** (1988), no. 1, 1–27.

[JG18] Daniel Jakubovitz and Raja Giryes, *Improving DNN robustness to adversarial attacks using jacobian regularization*, Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XII, 2018, pp. 525–541.

[KG92] J. Mark Keil and Carl A. Gutwin, *Classes of graphs which approximate the complete euclidean graph*, Discrete & Computational Geometry **7** (1992), no. 1, 13–28.

[KGB16] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio, *Adversarial examples in the physical world*, CoRR **abs/1607.02533** (2016).

[KH09] Alex Krizhevsky and Geoffrey Hinton, *Learning multiple layers of features from tiny images*, Tech. report, University of Toronto, 2009.

[KKG18] Harini Kannan, Alexey Kurakin, and Ian J. Goodfellow, *Adversarial logit pairing*, CoRR **abs/1803.06373** (2018).

[Kry84] Nikolai Vladimirovich Krylov, *Boundedly nonhomogeneous elliptic and parabolic equations in a domain*, Izvestiya: Mathematics **22** (1984), no. 1, 67–97.

[LCWC18] Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin, *Second-order adversarial attack and certifiable robustness*, CoRR **abs/1809.03113** (2018).

[LPV87] Pierre-Louis Lions, George Papanicolaou, and Srinivasa RS Varadhan, *Homogenization of Hamilton-Jacobi equations*, 1987.

[LYZ11] Songting Luo, Yifeng Yu, and Hongkai Zhao, *A new approximation for effective hamiltonians for homogenization of a class of hamilton–jacobi equations*, Multiscale Modeling & Simulation **9** (2011), no. 2, 711–734.

[MFUF18] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard, *Robustness via curvature regularization, and vice versa*, CoRR **abs/1811.09716** (2018).

[Mir14a] J. Mirebeau, *Anisotropic Fast-Marching on Cartesian Grids Using Lattice Basis Reduction*, SIAM Journal on Numerical Analysis **52** (2014), no. 4, 1573–1599.

[Mir14b] Jean-Marie Mirebeau, *Minimal Stencils for Monotony or Causality Preserving Discretizations of Anisotropic PDEs*.

[Mir16] ———, *Adaptive, anisotropic and hierarchical cones of discrete convex functions*, Numerische Mathematik **132** (2016), no. 4, 807–853.

[MMIK18] Takeru Miyato, Shin-ichi Maeda, Shin Ishii, and Masanori Koyama, *Virtual adversarial training: a regularization method for supervised and semi-supervised learning*, IEEE transactions on pattern analysis and machine intelligence (2018).

[MMS+17] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu, *Towards deep learning models resistant to adversarial attacks*, CoRR **abs/1706.06083** (2017).

[MRT18] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar, *Foundations of machine learning*, 2018.

[NBA+18] Roman Novak, Yasaman Bahri, Daniel A. Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein, *Sensitivity and generalization in neural networks: an empirical study*, 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, 2018.

[NK17] Vaishnavh Nagarajan and J. Zico Kolter, *Gradient descent GAN optimization is locally stable*, Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, 2017, pp. 5591–5600.

[NNZ19] R Nochetto, Dimitrios Ntogkas, and Wujun Zhang, *Two-scale method for the monge-ampère equation: Convergence to the viscosity solution*, Mathematics of Computation **88** (2019), no. 316, 637–664.

[NSZ17] Michael Neilan, Abner J. Salgado, and Wujun Zhang, *Numerical analysis of strongly nonlinear pdes*, Acta Numerica **26** (2017), 137–303.

[Obe05] Adam M. Oberman, *A convergent difference scheme for the infinity Laplacian: construction of absolutely minimizing Lipschitz extensions*, Math. Comp. **74** (2005), no. 251, 1217–1230 (electronic). MR MR2137000

[Obe06] ———, *Convergent difference schemes for degenerate elliptic and parabolic equations: Hamilton-Jacobi equations and free boundary problems*, SIAM J. Numer. Anal. **44** (2006), no. 2, 879–895 (electronic). MR MR2218974 (2007a:65173)

[Obe07] ———, *The convex envelope is the solution of a nonlinear obstacle problem*, Proc. Amer. Math. Soc. **135** (2007), no. 6, 1689–1694 (electronic). MR MR2286077

[Obe08a] ———, *Computing the convex envelope using a nonlinear partial differential equation*, Math. Models Methods Appl. Sci. **18** (2008), no. 5, 759–780.

[Obe08b] ———, *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), no. 1, 221–238.

[Obe13] ———, *Finite difference methods for the Infinity Laplace and p-Laplace equations*, Journal of Computational and Applied Mathematics **254** (2013), 65 – 80.

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

[OTV09] Adam M Oberman, Ryo Takei, and Alexander Vladimirsky, *Homogenization of metric Hamilton-Jacobi equations*, Multiscale Modeling & Simulation **8** (2009), no. 1, 269–295.

[OZ16] Adam M Oberman and Ian Zwiers, *Adaptive finite difference methods for nonlinear elliptic and parabolic partial differential equations with free boundaries*, Journal of Scientific Computing **68** (2016), no. 1, 231–251.

[PMG+17] Nicolas Papernot, Patrick D. McDaniel, Ian J. Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami, *Practical black-box attacks against machine learning*, Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, AsiaCCS 2017, Abu Dhabi, United Arab Emirates, April 2-6, 2017, 2017, pp. 506–519.

[PMJ+16] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami, *The limitations of deep learning in adversarial settings*, IEEE European Symposium on Security and Privacy, EuroS&P 2016, Saarbrücken, Germany, March 21-24, 2016, 2016, pp. 372–387.

[Pot17] Richard Potember, *Perspectives on research in artificial intelligence and artificial general intelligence relevant to DoD*, Tech. report, The MITRE Corporation McLean United States, 2017.

[PS04] Per-Olof Persson and Gilbert Strang, *A simple mesh generator in MATLAB*, SIAM review **46** (2004), no. 2, 329–345.

[RD18] Andrew Slavin Ross and Finale Doshi-Velez, *Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients*, Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, 2018, pp. 1660–1669.

[RL87] Peter J Rousseeuw and Annick M Leroy, *Robust regression and outlier detection*, vol. 1, Wiley Online Library, 1987.

[RLNH17] Kevin Roth, Aurélien Lucchi, Sebastian Nowozin, and Thomas Hofmann, *Stabilizing training of generative adversarial networks through regularization*, Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA, 2017, pp. 2015–2025.

[RLNH18] ———, *Adversarially robust training through structured gradient regularization*, CoRR **abs/1805.08736** (2018).

[RO94] Leonid I Rudin and Stanley Osher, *Total variation based image restoration with free local constraints*, Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference, vol. 1, IEEE, 1994, pp. 31–35.

[RSL18a] Aditi Raghunathan, Jacob Steinhardt, and Percy Liang, *Certified defenses against adversarial examples*, 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, 2018.

[RSL18b] Aditi Raghunathan, Jacob Steinhardt, and Percy S. Liang, *Semidefinite relaxations for certifying robustness to adversarial examples*, Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada., 2018, pp. 10900–10910.

[RVM+11] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio, *Contractive auto-encoders: Explicit invariance during feature extraction*, Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011, 2011, pp. 833–840.

[Sap06] Guillermo Sapiro, *Geometric partial differential equations and image analysis*, Cambridge university press, 2006.

[SBH] Andrew Shaw, Yaroslav Bulatov, and Jeremy Howard, *ImageNet in 18 minutes*.

[SLC19] Ismaïla Seck, Gaëlle Loosli, and Stephane Canu, *L1-norm double backpropagation adversarial defense*, arXiv preprint arXiv:1903.01715 (2019).

[SNG+19] Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John P. Dickerson, Christoph Studer, Larry S. Davis, Gavin Taylor, and Tom Goldstein, *Adversarial training for free!*, CoRR **abs/1904.12843** (2019).

[SOS+18] Carl-Johann Simon-Gabriel, Yann Ollivier, Bernhard Schölkopf, Léon Bottou, and David Lopez-Paz, *Adversarial vulnerability of neural networks increases with input dimension*, CoRR **abs/1802.01421** (2018).

[Stu99] Jos F Sturm, *Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones*, Optimization methods and software **11** (1999), no. 1-4, 625–653.

[SZS+13] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus, *Intriguing properties of neural networks*, CoRR **abs/1312.6199** (2013).

[TA77] Andrey N Tikhonov and Vasilii Iakkovlevich Arsenin, *Solutions of ill-posed problems*, vol. 14, Winston and Sons, New York, 1977.

[TKP+18] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel, *Ensemble adversarial training: Attacks and defenses*, International Conference on Learning Representations, 2018.

[TSE+18] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry, *Robustness may be at odds with accuracy*, CoRR **abs/1805.12152** (2018).

[TSS18] Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama, *Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks*, Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada., 2018, pp. 6542–6551.

[UOKvdO18] Jonathan Uesato, Brendan O'Donoghue, Pushmeet Kohli, and Aäron van den Oord, *Adversarial risk and the dangers of evaluating against weak attacks*, Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, 2018, pp. 5032–5041.

[Vil08] Cédric Villani, *Optimal transport: old and new*, vol. 338, Springer Science & Business Media, 2008.

[Wal45] Abraham Wald, *Statistical decision functions which minimize the maximum risk*, Annals of Mathematics (1945), 265–280.

[WK18] Eric Wong and J. Zico Kolter, *Provable defenses against adversarial examples via the convex outer adversarial polytope*, Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, 2018, pp. 5283–5292.

[WSMK18] Eric Wong, Frank R. Schmidt, Jan Hendrik Metzen, and J. Zico Kolter, *Scaling provable adversarial defenses*, Advances in Neural Information Processing

Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada., 2018, pp. 8410–8419.

[WZC+18] Tsui-Wei Weng, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, Dong Su, Yupeng Gao, Cho-Jui Hsieh, and Luca Daniel, *Evaluating the robustness of neural networks: An extreme value theory approach*, 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings, 2018.

[XGD+17] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He, *Aggregated residual transformations for deep neural networks*, 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, 2017, pp. 5987–5995.

[XTSM18] Kai Y. Xiao, Vincent Tjeng, Nur Muhammad Shafiullah, and Aleksander Madry, *Training for faster adversarial robustness verification via inducing relu stability*, CoRR **abs/1809.03008** (2018).

[XWvdM+18] Cihang Xie, Yuxin Wu, Laurens van der Maaten, Alan L. Yuille, and Kaiming He, *Feature denoising for improving adversarial robustness*, CoRR **abs/1812.03411** (2018).