Imperial College of Science, Technology and Medicine
Department of Computing

Optimal Sample Set Selection for the Simplex Gradient

Daniel Lengyel

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Computing of the Diploma of Imperial College, May 2024

Declaration of Originality

I Daniel Lengyel of Imperial College London hereby declare that this thesis and all presented material is my own work, except where specified explicitly. It does not contain material that has already been used for obtaining another degree. Due to the relatively long history and great breadth of research in computer science and engineering, it could be possible to find unintentional strong similarities with previous works, though I am not aware of such instances to the best of my knowledge.

Copyright

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC).

Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Acknowledgements

It is well known that the journey to a PhD is difficult and full of challenges. Many times one will wonder whether the choice was appropriate. I am also aware that it is a privileged position to be in. I do not take it for granted that I was allowed to continue my studies and pursue my passions. I am grateful to have had this opportunity, which often has felt selfish as it relies on the support of many.

I would like to express my sincerest gratitude to Nicholas R. Jennings. You have brought me in and your belief in me has never wavered. I want to thank Nikolas Kantas for your earnest feedback and the care you have put into my journey. Lastly, I want to thank Panos Parpas. You took me on as a student when I was looking for direction and I sincerely thank you for your guidance throughout these years.

I want to thank the computational optimization group, and in particular Ruth Misener, for extending my research group and creating a stimulating and incredibly supportive environment.

I want to thank my undergraduate mentors, who have been instrumental in my development. First, I want to thank Thanos Panagopoulos, you have championed my early research career and taught me how to preserve through difficult times. I want to thank Moustafa AbdelBaky, you have showed me how to become a much better software engineer, but more importantly how to be a good and kind person. Lastly, I want to thank Ananth Sridhar for being an endless source of curiosity and staple of what it means to dedicate oneself to a lifetime of learning.

Without the support of my friends I would not be where I am today. I am blessed to have more friends than I can mention, as leaving even one out would feel like a disservice. I am fortunate to have you all in my life and thank you deeply.

Lastly, I want to thank my family. Kathinka, I love you and can not thank you enough for having been by my side through all these years—I will forever be grateful. I want to thank my sister Kati, her husband Aleks and my niece Valentina for always providing a second home. I want to thank my brother Michael and his partner Oni, who have never ceased to keep me entertained when I needed it. Lastly, I want to thank my parents Margita and Jozef for enabling me to go on this journey. You have made the sacrifices so I could get the education that was not possible for you. I thank you and love you all.

Abstract

This thesis addresses the challenge of estimating the gradient of a potentially noisy black-box function. Accurate gradient estimation is critical in various domains, with applications ranging from parameter assessment of biological processes to trajectory control. In all of these applications, function evaluations are costly and should be minimized when possible. However, there remains only a restricted understanding of the optimal sample set—i.e., the ideal black-box evaluation locations. This understanding is limited to severely constrained structures or heuristic stochastic sampling strategies. This thesis aims to make progress toward filling this gap, with the goal of retaining accuracy while reducing the number of function evaluations, and hence minimizing the cost of gradient estimation.

We focus on the simplex gradient, the gradient of the best linear approximation of the black-box function. The simplex gradient is a robust and accurate estimator that generalizes many finite difference methods and can be seen as the deterministic counterpart to likelihood ratios. We focus on the two most common forms: the fully-determined simplex gradient, the generalization of forward differences, and the centered simplex gradient, a generalization of central differences.

For the fully-determined simplex gradient, Chapter 3 derives optimal, non-orthogonal sampling structures that exploit low-curvature regions, resulting in significant performance gains. The proposed CASG method achieves accuracy comparable to central differences at half the cost. Chapter 4 extends this work to the centered simplex gradient, connecting optimal sample sets to orthogonal roots of homogeneous polynomials. The ORCAS method developed here offers improved worst-case guarantees and shows numerical performance converging to theoretical optima, providing orders of magnitude improvement over existing methods.

Together, these contributions deepen our understanding of optimal sets for gradient estimation, offering practical methodologies to improve accuracy and reduce costs across diverse applications.

Contents

Declaration of Originality i

Copyright iii

Acknowledgements v

Abstract vii

Notation xix

1 Introduction 1

1.1 Motivation and Objectives 1

1.2 Contributions 5

1.3 Thesis Structure 8

1.4 Publications 10

2 Background Theory 12

2.1 Basic Gradient Estimation Problem 12

2.1.1 Direct Gradient Estimation Methods 14

2.1.2 General Gradient Estimation Methodology . . . 15
2.1.3 Motivation for our Choice on the Adaptive and Local Setting . . . 17
2.2 Properties of Gradient Estimation in Local Settings . . . 18
2.2.1 Function Stochasticity . . . 19
2.2.2 Function Structure . . . 20
2.2.3 Error Decomposition: Approximation and Noise Error Tradeoff . . . 22
2.3 Gradient Estimation Models . . . 24
2.3.1 Linear Models . . . 25
2.3.2 Non-linear models . . . 33
2.4 Applications of the Simplex Gradient . . . 34
2.4.1 Sensitivity Analysis . . . 34
2.4.2 Derivative Free Optimization and Meta Heuristic Algorithms . . . 34
2.5 Summary . . . 35

3 Optimal Sample Sets Selection for the Fully-Determined Simplex Gradient 37

3.1 Optimization Problem for Optimal Sample Set . . . 38
3.1.1 Approximation Error Bound . . . 39
3.1.2 Objective Function . . . 40
3.2 Theoretical Results . . . 41
3.2.1 Proof Outline of Theorem 9 . . . 44
3.2.2 Proof of Theorem 10 . . . 46
3.3 Error Rates and Geometry of Optimal Sample Sets . . . 53

3.3.1 Error Rates . . . 54
3.3.2 Geometry of Sample Set . . . 60
3.3.3 Example: Quadratic Function . . . 67
3.4 Algorithm: Curvature Aligned Simplex Gradient . . . 69
3.4.1 CASG: Base Algorithm . . . 70
3.4.2 Getting Λ^* . . . 71
3.4.3 Extending CASG . . . 73
3.4.4 Runtime . . . 76
3.5 Global Model Framework and Numerical Results . . . 76
3.5.1 Global Model Framework . . . 76
3.5.2 Sensitivity Analysis . . . 80
3.5.3 Derivative Free Optimization . . . 84
3.6 Proofs . . . 88
3.6.1 Poofs for Theorem 9 . . . 88
3.6.2 Proofs for Theorem 10 . . . 94
3.7 Summary . . . 99

4 Optimal Sample Sets Selection for the Centered Simplex Gradient 101

4.1 Background . . . 102
4.1.1 Centered Simplex Gradient . . . 102
4.1.2 Homogeneous Polynomials and Taylor's Theorem . . . 105
4.2 Optimization Problem for Optimal Sample Set . . . 106

4.2.1 Approximation Error Estimate via Polynomials . . . 108

4.2.2 Optimization Problem . . . 110

4.2.3 Solution when the difference matrix is orthogonal, Approximation and Noise error trade-off . . . 110

4.2.4 Case Study: Gradient Estimation . . . 112

4.3 Global minimum: d orthogonal roots do not always exist . . . 118

4.3.1 The Global Minimum: Orthogonal Roots . . . 118

4.3.2 Context and First Challenges . . . 120

4.3.3 There do not always exist d orthogonal roots . . . 122

4.4 ORCAS: Finding $d-1$ orthogonal roots . . . 130

4.4.1 Main Theorem and Proof . . . 130

4.4.2 ORCAS: Orthogonal Roots Centered (Almost) Simplex . . . 131

4.4.3 Finding Root . . . 132

4.4.4 Runtime . . . 135

4.4.5 Worst-Case Performance and Comparison to Central Differences . . . 135

4.5 Numerical Solvers . . . 138

4.5.1 BFGS as a Numerical Solver . . . 138

4.5.2 ORCAS as a Dimensionality Reduction Method . . . 139

4.6 Numerics: Exact Third Order Information . . . 141

4.6.1 Sampling random polynomials . . . 141

4.6.2 General Methodology . . . 145

4.6.3 Well-Conditioned Polynomials . . . 151
4.6.4 Ill-Conditioned Polynomials . . . 154
4.6.5 Fixed condition number . . . 156
4.6.6 Growing Condition Number with Dimension . . . 159
4.7 Numerics: Estimated Third Derivatives via a Global Model . . . 162
4.7.1 Global Model Framework . . . 164
4.7.2 Numerical Results: Ackley . . . 164
4.8 Proofs . . . 168
4.8.1 Extended Case Study Results . . . 169
4.8.2 Variance of ill-conditioned polynomial . . . 174
4.9 Summary . . . 179

5 Conclusions **181**

5.1 Summary of Thesis Achievements . . . 181
5.2 Applications . . . 183
5.2.1 Interacting Particle Methods . . . 183
5.2.2 Function Approximation . . . 184
5.3 Future Work . . . 185
5.3.1 Applications . . . 185
5.3.2 Historically Informed Sample Set Construction . . . 186
5.3.3 Further Study of Optimal Sample Sets . . . 187

Bibliography **188**

List of Figures

1.1 The structure of various simplex gradient instances and formulations. Dots representing sample points. 3

2.1 Visualizing the approximation and noise error tradeoff. Two samples of the black-box system are used to estimate the gradient at the given location via forward differences. 24

2.2 The structure of various simplex gradient instances and formulations. Dots representing sample points. 28

3.1 Comparing forward differences (FD) with optimal difference vector lengths [BFB15] to the curvature aligned simplex gradient (CASG). The first row corresponds to a positive definite Hessian with $k \in [10^{-4}, 10^{4}]$. The bottom row represents an indefinite Hessian with $k \in [-10^{4}, -10^{-4}]$ and with $h = 10^{2}$. 68

3.2 Comparison of gradient estimation methods for Ackley with $d = 8$ and $\sigma = 10^{-5}$. 82

3.3 Visualization of ODE path with three randomly chosen parameters around the recommended values. 83

3.4 Comparison of gradient estimation methods for sensitivity analysis of cell differentiation in the colon with $\sigma = 10^{-3}$. 83

3.5 Data profiles for L-BFGS using various gradient estimation methods with noise levels $10^{-1}, 10^{-3}$ and 10^{-5}. The top row are the profiles with $\tau = 10^{-1}$, associated with approximate convergence. The bottom row is $\tau = 10^{-5}$, under which the convergence criterion is only satisfied if a random run is able to get very close to the lowest achievable function value. 86

4.1 The structure of various simplex gradient instances and formulations. Dots representing sample points. 103

4.2 Visualization of the set of unit roots for two and three dimensional polynomials. For the three dimensional case we show the same set of roots but from perspectives rotated by some amount. 122

4.3 Wall Cock Time of Gradient Estimation methods. 140

4.4 Mean squared error as a function of dimension for the performance for well-conditioned polynomials in the performance regime $\tilde{h} \ll h$. 152

4.5 Mean squared error as a function of dimension for the performance for well-conditioned polynomials in the performance regime $\tilde{h} \approx h$. 153

4.6 Mean squared error as a function of dimension for the performance for well-conditioned polynomials in the performance regime $\tilde{h} \gg h$. 154

4.7 Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with a fixed condition number and in the performance regime $\tilde{h} \ll h$. 157

4.8 Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with a fixed condition number and in the performance regime $\tilde{h} \approx h$. 158

4.9 Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with a fixed condition number and in the performance regime $\tilde{h} \gg h$. 159

4.10 Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with an increasing condition number and in the performance regime $\tilde{h} \ll h$. 161

4.11 Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with an increasing condition number and in the performance regime $\tilde{h} \approx h$. 161

4.12 Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with an increasing condition number and in the performance regime $\tilde{h} \gg h$. 162

4.13 Mean squared error as a function of the number of sample points used to fit the global model for the Ackley function in dimension $d = 4$. 166

4.14 Mean squared error as a function of the number of sample points used to fit the global model for the Ackley function in dimension $d = 8$. 167

4.15 Mean squared error as a function of the number of sample points used to fit the global model for the Ackley function in dimension $d = 12$. 168

Notation

Symbol	Meaning
$\|x\|$	Euclidean vector norm
$B_\epsilon(x)$	Ball of radius $\epsilon > 0$ at x, $B_\epsilon(x) = \{z : \|z - x\| \leq \epsilon\}$
$\|A\|_2$	L_2 matrix norm
$\|A\|_F$	Frobenius norm
$\mathrm{tr}(\cdot)$	Trace operator
GL_d	General linear group of degree d, space of all invertible $d \times d$-matrices
$diag(v)$	Diagonal matrix with vector v along the diagonal ($v \in \mathbb{R}^d$, $diag(v) \in \mathbb{R}^{d \times d}$)
$diag(A)$	Diagonal of matrix A ($A \in \mathbb{R}^{d \times d}$, $diag(A) \in \mathbb{R}^d$)
$v \succ 0$	All entries of vector v greater than zero
$v \succeq 0$	All entries of vector v greater than or equal to zero
$B \succ 0$	Matrix B is positive definite
$B \succeq 0$	Matrix B is positive semi-definite
v_{max}	The largest element of v
v_{min}	The smallest element of v
D_{max}	When D is a diagonal matrix, the largest element on the diagonal
D_{min}	When D is a diagonal matrix, the largest element on the diagonal

Symbol	**Meaning**
D_i	When D is a diagonal matrix, the ith element along the diagonal
$\mathbb{R}_+$	Set of all strictly positive real values
$\mathbb{Z}_+$	Set of all positive integers
$\lvert\mathcal{A}\rvert$	Size of set $\mathcal{A}$
H	Hessian (second derivative of a function)
$H = RDR^T$	Eigen-decomposition of H, with R orthogonal and D diagonal
$S = U\Lambda V^T$	Singular value decomposition (SVD) of S, with U, V orthogonal and Λ diagonal given by $\Lambda_i = \sqrt{\lambda_i}$, $\lambda \in \mathbb{R}^d$ being the square of the diagonal of Λ
$f(x) = \mathcal{O}(g(x))$	Upper-bound on $f(x)$ such that there exists $c \in \mathbb{R}$ and subset $\mathcal{D} \in \mathbb{R}^d$ such that $f(x) \leq cg(x)$ for $x \in \mathcal{D}$
$f(x) = \Omega(g(x))$	Lower-bound on $f(x)$ such that there exists $c \in \mathbb{R}$ and subset $\mathcal{D} \in \mathbb{R}^d$ such that $f(x) \geq cg(x)$ for $x \in \mathcal{D}$
$f(x) = \Theta(g(x))$	Upper-bound and lower-bound on $f(x)$ such that there exists $c_1, c_2 \in \mathbb{R}$ and subset $\mathcal{D} \in \mathbb{R}^d$ such that $c_1 g(x) \leq f(x) \leq c_2 g(x)$ for $x \in \mathcal{D}$
$\arg\max_{x \in \mathcal{D}} f(x)$	The set of maximizers of f over the set $\mathcal{D}$
$\arg\min_{x \in \mathcal{D}} f(x)$	The set of minimizers of f over the set $\mathcal{D}$
M_k	Hadamard matrix, orthogonal with $(M_k)_{ij}^2 = \frac{1}{d}$ and all positive entries in the kth column
$\alpha = (\alpha_1, \ldots, \alpha_k)$	Multi-index notation, a k tuple with non-negative integers
$\lvert\alpha\rvert$	Order or degree of α, $\sum_{i=1}^d \alpha_i$
x^α	$x_1^{\alpha_1} \cdots x_d^{\alpha_k}$
$\alpha!$	$\Pi_{i=1}^k \alpha_i!$
∂_α	$\partial_{\alpha_1} \cdots \partial_{\alpha_k}$
$D^{\otimes k} f\vert_x[s]$	k-form, $\sum_{\lvert\alpha\rvert=k} \partial_\alpha \frac{k!}{\alpha!} f(x) s^\alpha$

Symbol	Meaning
$\partial_v^k f(x)$	If it exists, the k-th directional derivative of f at $x \in \mathbb{R}^d$ in the direction $v \in \mathbb{R}^d$, i.e. $\partial_v^k f(x) = D^{\otimes k} f\|_x[v]$
$R(\mathcal{A})$	Radius of $\mathcal{A}$, $\max_{1 \leq i \leq N} \|a_i\|_2$
$f(\mathcal{A})$	Function vector, component-wise $(f(\mathcal{A}))_i = f(a_i)$
$I_{d \times d}$	Identity matrix in $\mathbb{R}^{d \times d}$
A^{-T}	The inverse of the transpose, well-defined due to $A^{-T} = (A^T)^{-1} = (A^{-1})^T$
$\mathbf{1}_d$	A d-dimensional vector with all entries equal to one
$\mathcal{C}^k$	The class of k-times continuously differentiable functions

Chapter 1

Introduction

1.1 Motivation and Objectives

The gradient estimation problem focuses on estimating the gradient of a potentially noisy black-box function by only querying this function. This gradient represents the local change of a system with respect to the input variables. Being able to estimate this gradient precisely is crucial for a range of domains, two prominent ones being sensitivity analysis [VA00, KIT+19] and derivative free optimization [BBN19, BKW24, FH12], each with a range of applications. In sensitivity analysis the gradient is directly used to assess the change of an output with respect to some set of parameters. Here use cases range from studying the effect of parameters on biological processes [QM20, JEB+07, DPV06] to assessing system behaviour and critical inputs to financial and economic models [GLS00, Pan97]. In derivative free optimization (DFO) applications the gradient is used as part of an iterative process to find optimal parameters for a system. The applications of DFO span chemical product design [SSSC20], optimization of cardiovascular geometries [MFT08], pattern recognition and machine learning [FH75, MRFM20], and trajectory control and drone swarms [LK17, MCDDK16, KBD04]. An exciting aspect is that many at first seemingly unrelated domains have also been shown to have gradient estimation implicitly encoded, such as in meta-heuristic algorithms and interacting particle methods [STW23, BK98, CCSA20, DJR13, Oll18, Oll19]. The ubiquity of gradient estimation methods

and its range of applications hence underscores its impact and importance of study.

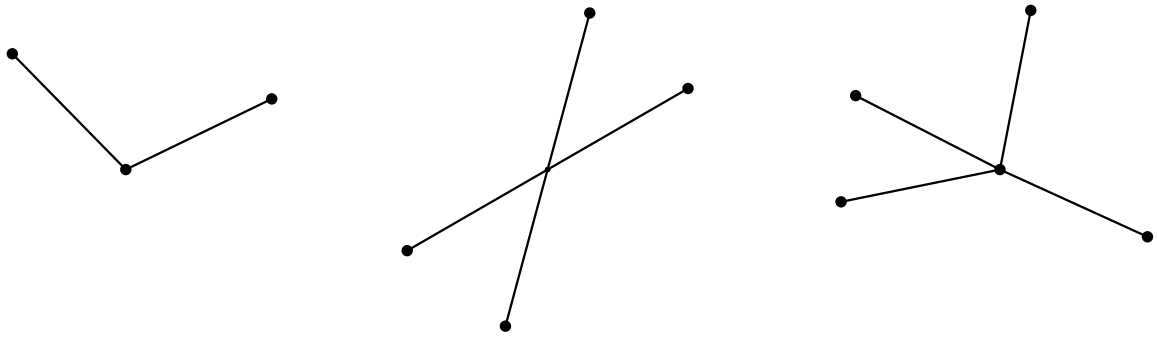
The key limitation of gradient estimation is that evaluating the system is assumed to be expensive and should be limited when possible [MRFM20, Fu06, LMW19, BBN19]. This is a natural constraint, as most systems where gradients are not immediately available involve complex simulations and physical processes [LK17, MCDDK16, KBD04, GLS00, Pan97]. Since the gradient estimation methods are constructed via information gained from querying the system, these evaluations have to be done carefully to reduce this cost. Yet, for the most common gradient estimators there is only a limited and highly constrained understanding of where the function should be evaluated to achieve the most accurate gradient estimates [BBN19, LMW19, HJB20, HJBP22, Reg15]. This is a major limitation, as without such an understanding many more function evaluations than necessary will be used, significantly increasing the cost in a range of applications.

To frame the contributions of this thesis, we provide a brief description of the gradient estimation problem. We focus on a setting where one needs to choose a sample set, that is the locations at which to evaluate the black-box system as part of the gradient estimation problem. This is in contrast to methods which purely rely on historical data for the gradient estimate. While in practice one would want to use all the data available, focusing on the sample set allows for a principled theoretical study of what an ideal dataset may look like. Furthermore, we focus on local methods, that is using data that is obtained close to the gradient estimation location. This setup allows us to require less prior information on the function structure. We note that while such information is often not present in practice, when more information is available other assumptions may be appropriate.

Given our setup, we focus on the simplex gradient, which is the gradient of the best linear estimate of the function. This is a strong estimator to study, since it describes the structure of the function locally, which is in-line with the assumptions of local sampling. Due to its linear structure it is also a robust gradient estimator in the presence of noise. Lastly, the simplex gradient is a generalization of many numerical differentiation methods and occurs in a variety of application areas as we discuss in Chapter 2. We note that in other settings, where more

function evaluations may be available and noise is not present, more complex estimators may be useful.

Depending on the choice and geometry of this sample set, the simplex gradient will have vastly different approximation properties and performance. We present the two most common ones in Figure 1.1, which will also be the ones studied in the thesis. Even though these sample sets provide significant freedom in the selection of sample points, this freedom has not yet been explored and commonly only restrictive geometries, such as orthogonal ones, have been assumed [HJBP22, CH23, SXXN22, CSV08b]. Hence, *our core objective will be the study of the optimal sample set for the simplex gradient.*



(a) Fully Determined Simplex Gradient (b) Centered Simplex Gradient (c) Simplex Gradient

Figure 1.1: The structure of various simplex gradient instances and formulations. Dots representing sample points.

The challenge in choosing an appropriate sample set is the noise-approximation error tradeoff. The closer the points are the better the approximation error becomes. However, the error due to noise increases due to increased variance in the estimates. We provide a closer look at this problem in Section 2.2.3. The important insight is that depending on the placement of the points, the gradient estimator may have significantly improved error qualities. For this note that in some directions, the gradient changes more slowly. When placing sample points along such directions, the distance between sample points can be increased to subsequently decrease the noise error.

To know where to place the sample points, the function needs to be understood. The benefit of

using a simplex gradient is that locally most functions can be well approximated by a polynomial as given by Taylor's Theorem. So, instead of having to derive an optimal sample set for various different function classes, it often suffices to only consider low-degree polynomials as they are able to explain most of the local variance. An added benefit is that polynomials are well understood and lead to increased interpret-ability in the ultimate choice of the sample set. However, since the polynomial expansion generally uses no further assumption on the function class and focuses on local approximations, if further information was available, other expansions may also be appropriate.

Once a polynomial expansion is used, the description of finding an optimal sample set has a significantly simplified interpretation. Specifically, in the case of the fully-determined simplex gradient the problem reduces to finding the optimal sample set for a quadratic function. When using a centered simplex gradient the problem reduces to a third-order polynomial. This is because the geometry of the centered simplex gradient fully removes the second order error, but at the expense of twice the number of function evaluations.

While the problem becomes interpretable, there remains good reason for a lack of answers to the optimal sample set problem. The study of the sample set is a bit of a chicken-and-egg problem. To study the optimal sample set we need to know the function. But if we knew the function we would not need to perform any gradient estimation in the first place.

A different perspective on this dilemma makes the problem approachable. Rather than immediately developing a practical method, we first study what theoretic characteristics make a good sample set and what properties of the function may inform their optimal choice. For this, we simply assume to know the function exactly. Methods can then be designed to estimate function-dependent properties in anticipation of using them to inform the right sample set. Some methods have already used such approaches by further sampling the system [GMW19, GMSW83, SXXN22, MW12, Bar92]. The downside is that they are expensive, do not use any potential historical knowledge, which may be available in settings such as optimization or sensitivity analysis, and provide only limited structural information. Hence, *an additional objective will be to address these practical shortcomings in estimating function-dependent infor-*

mation.

1.2 Contributions

The following contributions for the simplex gradient sample set are derived by assuming exact higher-order information of the function. These results remain valuable for a theoretical study of good sample set structures. However, as this information is often not available in practice, we provide the following contribution to prove practical feasibility and address the shortcomings of other developed methods.

- We introduce the historically informed global model framework for sample set selection. The method is based on using past function evaluations to fit a surrogate model that provides a rough sense of the function locally. We can then obtain any higher-order estimates needed for the sample set selection. This is in contrast to current methods which rely on further function evaluations and do not explicitly fit a global model to obtain the required function-dependent information. In Section 3.5.3 we show that our method outperforms the current state of the art methods used to estimate function-dependent information without having to further query the system. Furthermore, in Sections 3.5 and 4.7 we demonstrate its efficacy in conjunction with our developed sample set selection methodologies where other methods are not applicable.

The above contribution both justifies the development of optimal sample sets assuming knowledge of higher-order information and provides a practical framework within which to employ such methods.

We now present our results from Chapter 3 which focus on the fully-determined simplex gradient for functions which are at least twice differentiable. Here only $d + 1$ sample points are used, with the only restriction being that the point at which the gradient is to be estimated has to be included in the sample set.

- We derive the optimal sample set for a fully-determined simplex gradient on any quadratic function in Theorems 9 and 10. To account for higher-order errors, the result also allows for constraints on the size of the sample set. This represents the first optimality result for the simplex gradient which does not impose any constraints on the sample set.

- We provide an analysis of the error rates and geometry of the optimal sample set in Section 3.3. Our results show a significant improvement over the standard finite difference errors. Regarding the geometry, we learn that in some cases the optimal sample set is able to fully remove the approximation error. In the remaining instances, the sample set exploits low curvature directions to improve the performance of the gradient estimator. Both insights are in contrast to finite difference methods. Furthermore, they provide a strong argument against the common wisdom that orthogonal sample set structures should be preferred. Overall, these results highlight key properties that can be used to guide the structuring of sample sets in order to improve performance in practice.

- We introduce the curvature aligned simplex gradient (`CASG`) which is based on our theoretical results in Theorems 9 and 10 and the insights gained in Section 3.3. `CASG` also provides a heuristic extension to arbitrary dimensions, to account for the limitation of Theorem 9 being valid only for dimensions which are multiples of four [1]. `CASG` heuristically partitions the space and we prove that under this partition `CASG` is the optimal sample set for the full-determined simplex gradient in Proposition 33.

- We numerically evaluate CASG and show that this sample set derived for a second order approximation of a function improves orders of magnitude on any comparable gradient estimator. We also evaluate our method via the historically informed sample set selection framework with similar results. We crucially find to perform similarly to central differences, the de facto standard, while using only half the number of function evaluations.

Our findings in Chapter 3 hence make a significant step toward understanding optimal sample

[1] In fact, the Theorem is true for all dimensions for which the Hadamard matrix exits. The current conjecture is that this is true for all dimensions which are multiples of four. Currently, the lowest multiple of four dimension for which no Hadamard matrix has yet been found is 668 [KTR05] — a dimension too large to worry about in many practical applications.

set structures in the case when only very few function evaluations are available. As functions are locally well approximated by a quadratic structure, finding an optimal sample set in this context directly tackles the bottleneck error encountered by any fully-determined sample set. *Our finding further implies that in many situations we can replace central difference methods, the de facto standard, with limited loss in performance while reducing the cost by a factor of two.*

In Chapter 4 we focus on the centered simplex gradient for functions which are at least three time differentiable. Here, $2d$ points are used with a symmetric layout. This structure removes the second order error, by removing all even order terms, and a third-order polynomial explains most of the local variance in the function. For this problem, we have made the following contributions.

- For the centered simplex gradient we show that the problem reduces to finding orthogonal roots for a homogeneous third order polynomial. We prove that for any third degree homogeneous polynomial there exist $d-1$ orthogonal roots in Theorem 75 [2]. We prove that the above result is tight. Specifically, we prove that there are third-order polynomials for which no complete orthogonal set of roots exist in Corollary 73. This represents a novel result on the structure of roots for homogeneous polynomials and is the first to link them to the optimality of the sample set.

- Based on our results we develop `ORCAS`. The method finds a complete sample set by finding $d-1$ orthogonal roots and then appending a final orthogonal vector to this set. We prove that as the dimension increases this approaches a best case performance. We also show that `ORCAS` has significantly better worse case guarantees than the best central difference method. These results are presented in Section 4.4.2. We again use a constraint on the sample set size to account for higher-order errors. Our method is the first to provide such worst-case guarantees.

- We show that `ORCAS` can be used as a dimensionality reduction method employed in

[2] This result also holds for all odd degree homogeneous polynomials, which may be of interest as a more general result.

conjunction with a numerical solver. Specifically, `ORCAS` reduces the problem to a dimension in which a numerical solver can find an optimal sample set. Without the dimensionality reduction, the run time of the numerical solver would be prohibitive. We present the construction in Section 4.5. These methods provide a further practical alternative and allow a practitioner to select among a range of principled sample set selection methods. They do so by addressing the constraint of `ORCAS`, by allowing the last few sample points to break the orthogonality constraint to allow for a potentially lower approximation error.

- We provide extensive numerical results and a broad study of the gradient estimation problem within a case study in Section 4.2.4. We show empirically that the improvement gain of a method follows the best case improvement which we show to be orders of magnitudes better than existing methods, as seen in Section 4.6.

- We evaluate the method within the historically informed sample set selection framework in Section 4.7. We show up to a three to four fold improvement when using our developed methods to current methods.

Our methods lead to two important realizations. While for the fully-determined simplex gradient the conventional orthogonal sample set structure was often not optimal, in the centered simplex gradient setting this structure is often beneficial. This is due to *our result that there exist many orthogonal roots, implying that a near optimal solution is possible with an orthogonal sample set geometry.* Hence, there lies significant use in this conventional structure. So, *a careful layout can lead to multiple magnitudes of improvement when employed correctly.*

1.3 Thesis Structure

In Chapter 2 we provide the necessary background and set-up to fully define and address the gradient estimation problem. We first define the problem and introduce our assumptions. We then provide a review of gradient estimation methods, with a specific focus on the simplex gradient being a generalization of many numerical differentiation methods. We conclude with a

review of the applications of gradient estimation methods and highlight the close connection to simplex gradient type estimators. Throughout this chapter we highlight the lack of a principled sampled set and data-selection methodology for the simplex gradient. We argue that due to the simplex gradient being a crucial component in a range of applications and domains, addressing this lack of understanding is important as it promises cost-savings and increased accuracy. The following Chapters aim to make steps towards filling this gap in our knowledge.

In Chapter 3 we address the optimal sample set structure of the fully-determined simplex gradient. This case covers the most basic setting of the simplex gradient, using the fewest number of sample points possible. It is important to address this case to study the behaviour of the simplex gradient when only a small budget is available. In Theorem 9 we then present the optimality conditions of the sample set, representing the basis of the curvature aligned simplex gradient (`CASG`). We also discuss the intuition behind the optimality conditions, a crucial takeaway to construct other methods based on our optimality results. We then provide extensive numerics showing that both in a theoretical optimal case and in practical applications, via the use of the historically informed global model framework, `CASG` is able to provide orders of magnitude improvements over contemporary methods. Crucially, we show similar performance to central differences, the de facto standard, while using only half the number of function evaluations. The caveat of this method is that if a budget of twice the number of function evaluations is available, any method based on a centered simplex gradient will be able to achieve similar results as `CASG`. This is due to a centered method due to its structure fully removing the approximation error which `CASG` is addressing.

In Chapter 4 we then focus on the centered simplex gradient, to address the case when twice the number of function evaluations are available. This problem is crucial when a higher budget is available and further structures can be exploited to minimize the error in the gradient estimate. Due to the centered simplex gradient, the gradient estimation problem differs to the one in Chapter 3 and is significantly more involved. Hence, new methodologies and approaches have to be developed. Therefore, this Chapter represents a deep study of centered simplex gradient methods, with a range of results to assess its best-case performance and how these relate to the specific function structure. The main result of the Chapter is based on a strong

connection between optimal sample sets and orthogonal roots of third order homogeneous polynomials. Based on this connection, we develop various results to develop a method, `ORCAS`, which approaches the best-case performance of any centered simplex gradient estimator with an increasing dimension. We also show that `ORCAS` attains orders of magnitude performance gains over currently used centered simplex methods. We then introduce further methods based on `ORCAS` to alleviate some of the constraints placed on the sample set. We conclude with a range of numerical experiments to show the improvement gains of our methodology. With this Chapter, we have studied the most used simplex gradient setting and provided a range of results. While the results are promising, we did not provide explicit optimality conditions as in Chapter 3, which may be addressed in future work.

We then conclude our work in Chapter 5. We first present a summary of the thesis achievements. We then give a more holistic view on possible future applications. For this, we summarize application domains we have found to be closely connected to the simplex gradient but have not seen more explicit mention of. We then finish with a look forward on potential future work. Here we consider work to either extend our methods to specific application domains or strengthen the developed results.

1.4 Publications

Our main work has focused on the gradient estimation problem. One publication has received Major Corrections form IMA Numerical Analysis. The work which compromises the second Chapter is currently in preparations.

- **D. Lengyel**, N. Kantas, P. Parpas, N. R. Jennings. Curvature Aligned Simplex Gradient (CASG): Optimal Sample Set Selection for Efficient Gradient Estimation, Major Revision at IMA Numerical Analysis.
- **D. Lengyel**, N. Kantas, P. Parpas, N. R. Jennings. Orthogonal Roots: Towards the Optimal Sample Set for the Centered Simplex Gradient, In Preparations.

To keep updated with the evolving community outside of my field I have worked on a variety of other topics.

- R. Pandya, P. Parpas, **D. Lengyel**. From Plateaus to Progress: Unveiling Training Dynamics of PINNs, NeurIPS Machine Learning for Natural Sciences workshop, 2023.
- **D. Lengyel**, A. Borovykh. Efficient regression with deep neural networks: how many datapoints do we need?, NeurIPS Has it Trained Yet? workshop, 2022.
- **D. Lengyel**, J. Petangoda, I. Falk, K. Highnam, M. Lazarou, A. Kolbeinsson, M. Peter Deisenroth, N. R. Jennings. GENNI: Visualising the Geometry of Equivalences for Neural Network Identifiability, NeurIPS Differential Geometry meets Deep Learning workshop, 2020.

Chapter 2

Background Theory

In this chapter, we lay out the foundation and context for the gradient estimation problem. We begin with Section 2.1 and 2.2 by defining the problem and the *local* and *adaptive* gradient estimation setting which we will discuss in this thesis. We then discuss the various gradient estimators that are available for this setting in Section 2.3. We put specific emphasis on linear models and specifically highlight the relationships between them, showing that the simplex gradient is a generalization of a range of gradient estimation methods. Lastly, we discuss various application areas of gradient estimation models, with a specific focus on highlighting the use of the simplex gradient in Section 2.4. Here we want to argue the ubiquitousness of such models and hence the impact a more principled sample set selection method can have.

2.1 Basic Gradient Estimation Problem

The core problem we aim to solve is *estimate a gradient using only function evaluations, using as few evaluations as possible*. We expand on this briefly and lay out our assumptions. Then we describe the general methodology and classifications of this problem, in the process highlighting our focus.

We base the expansion of the general problem on Fu et al. and their description of indirect gradient estimation methods [Fu06]. For this, we define a black-box function as a random

process to formalize the notion of sequentially evaluating a black-box. The black-box is given by $\tilde{f} : \mathcal{D} \times \mathbb{Z}_+$ where $\mathcal{D} \subseteq \mathbb{R}^d$ is an open set. We make the following assumptions.

1. $\tilde{f}(x, i)$ is a random variable, with $\tilde{f}(x, i)$ and $\tilde{f}(x, j)$ i.i.d. for $i \neq j$,
2. $\mathbb{E}[\tilde{f}(x, i)] = f(x)$ for all $i \in \mathbb{Z}_+$ and $f \in \mathcal{C}^1$,
3. $\tilde{f}(x, i)$ has bounded second moment.

The second assumption is natural, as we will be estimating the mean gradient and hence f. Although the other two assumptions appear in most texts focused on gradient estimation, they could be relaxed for more general results in future work [Fu06, Sch22, BH12, BDHdH08]. These assumptions then directly lead to writing $\tilde{f}(x, i) = f(x) + \epsilon(i)$, where $\epsilon(i)$ is a mean zero random variable with bounded variance and with $\epsilon(i), \epsilon(j)$ i.i.d. for $i \neq j$. As above, we naturally assume that f is at least differentiable, with stricter conditions listed as needed.

The key assumption is that $\tilde{f}$ is generally expensive to evaluate. This is of course not a rigorous statement, and the exact cost of generating function evaluations needs to be taken into account on a per-case basis. Fu et al. lists a range of reasons, such as each individual sample being expensive to generate or having to generate a lot of samples of $\tilde{f}$ to obtain an approximation of f to the desired accuracy. It may also be that obtaining a sample takes a long time and is not easily computed [Fu06].

The description of costly and noisy function evaluations is a key assumption. Otherwise, the problem will become near trivial to solve. If we could use arbitrarily many function evaluations, then via simple Monte-Carlo techniques we can increase the precision of the estimate of f to any desired level. The problem then becomes simple to solve, as even the most basic numerical differentiation schemes, such as finite differences, will be appropriate [BH12]. However, in many cases, the system is both corrupted and difficult to evaluate.

To demonstrate the applicability of the above description, consider assessing the sensitivity of the control parameters in a chemical process [MOV12, SSSC20]. Via a gradient estimate, we know how the function changes locally and can then improve the performance by moving

the parameters appropriately. Usually, the chemical process is a system that cannot be simply mathematically modeled, we may only be able to interact with it via querying the system. However, each query of the system will likely be slightly corrupted by noise, due to unexplained factors impacting the system. Hence, we may model it via the process $\tilde{f}$. Since querying the systems requires executing a complete chemical process, and each run may additionally be costly, as few function evaluations as possible are preferred.

The goal of gradient estimation is then *providing an accurate estimate of* $\nabla f(x)$*, while only using function evaluations of a black-box system* $\tilde{f}$*, while minimizing cost.* Some aspects remain open. For example, we need to cover what an accurate estimate means and what access to data looks like. First, we contrast the above setup to ones where direct methods, such as automatic differentiation, can be employed.

2.1.1 Direct Gradient Estimation Methods

In our above description, we enforced the condition that gradient estimation should only consist of function evaluations directly obtained from $\tilde{f}$. This requirement consolidated the description of Fu et al. of indirect gradient methods. Specifically, no knowledge of the underlying mechanics of f is used to derive the gradient estimate and no changes are made in the execution of the system itself [Fu06]. Otherwise, if these requirements were violated, other methods may be more beneficial.

Fu et al. describe estimators that do not fall under this category as direct gradient estimators. Such methods try to "estimate the true gradient using some additional analysis of the underlying stochastics of the model". When simple source code is available, then methods such as automatic differentiation are the first choice [G+89, BPRS18, SHWA15]. In other cases, methods that use domain knowledge are used, such as infinitesimal perturbation analysis [Sur87, WCC18], score function estimators [Wil92] or weak derivative estimators [BKK19].

Direct estimators' advantages include oftentimes unbiased gradient estimates, eliminating the need to determine hyper-parameters of indirect methods, and computationally more efficient

methods. In cases when direct estimators are available, their main downside, compared to indirect methods, is that they are slower and more difficult to implement. Direct methods require offline-work, problem-specific analysis, and modifications to how simulations are carried out.

Still, there are many instances where direct methods cannot be employed. These included systems where mathematical models are not available, or are difficult to obtain, or legacy code that cannot be modified. Hence, since such methods are employed precisely in situations when systems are difficult to interpret and expensive to evaluate, understanding indirect methods better is crucial to make gradient estimates viable.

2.1.2 General Gradient Estimation Methodology

We now describe the general methodology and components of gradient estimation and provide an appropriate classification. This will allow us to better describe our approach and assumptions of the problem. The following will be based on a range of works as we could not find an appropriate summary.

We identify the following four core components used as part of a gradient estimation problem.

- A *evaluation history* $\mathcal{E}$ which is made up of historical function evaluations. Specifically, let j be the current time and letting I be an index set such that $i < j$ for $i \in I$, the evaluation history is given by $\mathcal{E}_I = \{(x_i, \tilde{f}(x_i, i)\}_{i \in I}$. Here, x_i are the locations of the historical evaluations and $\tilde{f}(x_i, i)$ are the realized observations.
- A *sample set* given by $\mathcal{X}$. The sample set specifies the location of function evaluations that should be performed for the use of a gradient estimate. The size of the sample set should often be as small as possible, since function evaluations are assumed to be expensive.
- The *gradient model* given by $\mathcal{G}$. The gradient models uses the evaluation history and the sample set to construct a gradient.
- A *loss function* to assess the performance of the gradient estimator.

The first two components describe the data acquisition. They make precise that in any setting we have access to some past function evaluations, and as part of the gradient estimate we may conduct further queries [AH17, HJBP22, BBN19, AA07, BDHH05]. The third component focuses on the model which constructs the gradient estimate. The structure of the gradient model is often based on fitting a function $\mathcal{F}$ to the data available. We present a range of such models in Section 2.3. With the last component we make the role of the loss function concrete. While this could simply be seen as an aspect of the model, we emphasise that the choice of the loss will greatly influence the type of model and sample set chosen. In our work, we focus on a mean squared error loss, as it assess both the direction and the magnitude of the gradient estimate and is the most commonly used measure [BBN19, BH12, HJBP22]. However, if it is more important to obtain the correct direction, as may be the case in exact-line search optimization methods, then a loss such as the cosine similarity may be appropriate [NW06, SY06].

For a practitioner, the choices to be made often focus on the sample set and gradient model as both the evaluation history and loss will generally be fixed depending on the application. Of the two flexible components, the sample set is usually the one which has most constraints imposed on it. Based on these constraints, the choice of a gradient model generally follows naturally. We now briefly categorize the various choices made around the sample set and discuss their uses. We will then expand on the choices we have made for the thesis.

We first distinguish between a fully adaptive and fully historic setting. While some work has touched on this distinction for an application of gradient estimation in graphics [SPD18], the distinction is mostly in the vein of other adaptive methods, especially found in the design of experiments literature [FFN21, LOC18, BPL19, LBXY19]. We describe a fully *adaptive setting* as one where the gradient model only uses the results from the sample set for gradient estimation. Crucially, in this setting the sample set selection itself may be informed by the evaluation history, but the model itself does not receive any data from it. In contrast, a fully *historic setting* is one where only past data is used to construct a gradient estimate. Here no more further data needs to be sampled. The middle ground is using adaptive sampling to complement the historic data. When appropriately done, this may in practice yield optimal results as all data is utilized.

The other distinction focuses on a fully local and fully global setting, which follows the spirit of Ahnert et al. [AA07]. A *local* setting uses data which is close to the location of gradient estimation. In the local setting, a function can often be well expressed by polynomials, as justified by Taylor's Theorem. On the other hand, a *global* data strategy focuses on utilizing data that may be spread out. Here often some knowledge of the general function class into which the system belongs may be necessary as simple structures, such as polynomials, will be poor approximators. However, in the global setting more data, possible stemming from the evaluation set, can be employed.

2.1.3 Motivation for our Choice on the Adaptive and Local Setting

For this thesis we focus on the *adaptive* and *local* setting. Here we consider gradient estimation methods which construct gradient based fully on the evaluations from the sample set. The sample set is also constructed to focus on local function evaluations.

The benefit of a local method is that we can work in a setting where we can use a simple representation of the problem, namely polynomials, and simplify our assumptions on the noise structure of the system $\tilde{f}$. We expand on these properties in more detail in Section 2.2. In summary, in a local setting, we can treat the variance of noise to be close to constant. Furthermore, we know that the lowest order polynomials explain most of the variance locally, making it efficient in terms of the number of expansion terms needed to obtain strong approximations. The natural downside is that we can not yet efficiently incorporate global data into fitting the gradient model. Hence, a lot of historical data may then be discarded. Future work may focus on how to incorporate global data. We believe that Mutny et al. have set an interesting direction where they represent the problem in a reproducing kernel Hilbert space, where a kernel-based expansion of the function can be used [MK22].

Focusing on an adaptive setting allows us to better understand the impact of the data on gradient estimation. In the adaptive setting, we assume that for every gradient estimate, a new sample set is constructed. While in practical applications, previously evaluated locations may be included in the sample set to reduce cost, in the adaptive setting, we are interested in the

theoretically optimal choice of the sample set. This will later inform which previously evaluated locations may be good to include or discard in future sample sets.

Historically Informed Sample Set Selection

While in the adaptive setting, the data used by the gradient model is assumed to come from the sample set, the sample set selection often requires some estimate of function-dependent properties, as otherwise it cannot be adapted to a specific problem. Although some methods have focused on estimating these function-dependent properties, they have generally further sampled the system to obtain these and hence incurred further cost [GMW19, GMSW83, SXXN22, MW12, Bar92]. Even though these methods attempt to reduce the number of function evaluations by estimating the higher-order information only periodically, the procedure remains expensive as more function evaluations than strictly needed for gradient estimation are required. Therefore, there remains significant room for also incorporating the historical sample set, as using it will not incur further cost.

2.2 Properties of Gradient Estimation in Local Settings

We now discuss some of the key properties and assumptions when working in an adaptive and local gradient estimation setting. For this, we first discuss the noise structure. We use this to justify our choice in our contributions to focus on cases with constant variance noise. Then we consider the local function structure. Here we present the relevant results for approximating functions locally. Specifically, we introduce the notation around polynomial expansions. Then, we discuss the key difficulty in this setting that arises from the approximation and noise error trade-off. An understanding of this problem greatly aids in the discussion of gradient estimation methods for this setting.

2.2.1 Function Stochasticity

In the general setting, the function is represented simply as a random process $\tilde{f}$ as given above. When considering a global setting, there needs to be a strong understanding of the structure of the noise. Some works have focused specifically on this by considering RKHS settings [MK22]. However, in a local setting, many fewer assumptions are necessary on the random process to obtain strong approximations. While these assumptions will not hold true for all problems, we will heuristically argue that for many problems it is reasonable that the following assumptions are sensible. Furthermore, these assumptions are common in the literature as they cover many practical applications [CSV08a, BCCS19, BBN19, BH12].

Let us write the function structure as follows

$$\tilde{f}(x, i) = f(x) + (\tilde{f}(x, i) - f(x))$$

where by assumption $\tilde{f}(x, i) - f(x)$ is a random variable with bounded second moment and zero mean. We can hence think of the noise as being additive. However, the variance of this noise component may vary at different locations. We now heuristically argue that due to locality, the variance will vary little in the region of the sample set. Specifically, we want to argue that for every pair of elements of the sample set, i.e. $x, x' \in \mathcal{X}$, we have that $\mathbb{V}[\tilde{f}(x, i) - f(x)] \approx \mathbb{V}[\tilde{f}(x', i) - f(x')]$.

The reasoning is based on representing the variance of the random process as a function of x and that if x and x' are sufficiently close together, and the function is well-behaved with a "sensible" Lipschitz-constant, the variance should not vary too much locally. A common example where the noise varies based on the location is given by geometric type noise. This type of system is written as $\tilde{f}(x; i) = f(x)(1 + \epsilon)$ where ϵ is a random variable with zero mean and variance σ^2. Clearly, the strength of the stochasticity will vary directly proportional to the value of $f(x)$, as the variance is given by $f(x)^2\sigma^2$. When the function does not vary much then the variance will vary little also.

The above will not hold for all problems, as we can always construct functions for which the

gradient is high and hence even a small deviation from a point will lead to a great change in variance. However, we have found our assumption of constant variance to cover many practical applications and is a common assumption made in the literature [BBN19, CSV08a, CSV08b, BH12, AA07]. We make this assumption precise in the following.

Assumption 1. *Let $\tilde{f}(x, i) = f(x) + \epsilon(i)$. We assume that $\epsilon(i)$ has zero mean and constant bounded second moment. We additionally assume that for $i \neq j$, $\epsilon(i)$ is independent of $\epsilon(j)$ and that $x(i)$ is independent of $\epsilon(i)$.*

Note that we additionally make an implicit assumption explicit, namely that subsequent function evaluations have independent noise. The last assumption that $\epsilon(i)$ is independent of $x(i)$ makes explicit that we cannot choose x by predicting the noise.

While for the purposes of our results we assume constant variance, we believe that there is room to further investigate the role of different noise structures. Some directions may include investigating the sensitivity of any sample set constructed to the variance used. Then, we may gain an understanding of how much a more correct representation of local noise may aid performance. For settings in which the noise component plays a central role, then a method which is more granular and incorporates noise more accurately will benefit.

2.2.2 Function Structure

To select good data to be used for gradient estimation, some understanding of the function structure is needed. This is a delicate dance: If we knew the whole function structure, then we would not need to estimate gradients. We would know them exactly. Hence, the correct assumptions have to be made such that a practical method may hope to capture the required information about the function structure.

In global settings, a strong prior on the function is required. This is needed to understand the behavior of the function between datapoints and infer the structure through that lens. An example of this is by assuming the function is given by some kernel structure, or that the

black-box system $\tilde{f}$ is a Gaussian Process [MK22]. Here, then each sample of the Gaussian Process should have some understood properties. However, the difficulty is that we would need to have a good guess on the prior and learn the function structure in the process. Often, we do not have this luxury.

In a local setting, many difficulties in choosing a prior are removed. Locally, we know that most of the variance of the function is explained by lower order polynomials. Specifically, by Taylor's Theorem, we know that to approximate the function locally well, only the lowest order polynomials are required. Now, in practice we will not know the higher-order information. However, we are argue that even very rough information will be helpful. As we have argued earlier, a local adaptive gradient estimator should always improve on a global model. So, using it to bias the sample set seems sensible.

Representing Functions as Polynomials

We will be working a lot with this set-up, so we also introduce all relevant notation here. Specifically, we will think of the function as a polynomial from now on, with the lowest order being addressed for all problems.

Definition 1 (Homogeneous Polynomial)**.** *A d dimensional homogeneous polynomial of order k is a polynomial $p : \mathbb{R}^d \to \mathbb{R}$ with the property that for $\lambda \in \mathbb{R}$ and any $x \in \mathbb{R}^d$ we have $p(\lambda x) = \lambda^k p(x)$.*

An example of a homogeneous polynomial in four variables of order two is $p(x_1, x_2, x_3, x_4) = x_1x_2 + x_2x_3 + x_3x_4$. Another example is the quadratic form induced by the Hessian of a function.

We will not make the distinction between a form and a homogeneous polynomial and may use them interchangeably. In some texts a polynomial refers to an expression in terms of an indeterminate and a form denotes the induced function. Since we only consider the field of real numbers, the form and homogeneous polynomial are used interchangeably.

To see the intricate connection between homogeneous polynomials and the functions induced by the higher-order derivatives we note the following. The kth derivative of a k-times differentiable

function f defines a form via $D^{\otimes k} f|_x[s]$. Conversely, for every homogeneous polynomial of order k there exists a function f such that $D^{\otimes k} f|_x[s] = p(s)$ with the trivial example being $f(x) = \sum_{|\alpha|=k} \frac{\alpha!}{k!} c_\alpha x^\alpha$ which gives $D^{\otimes k} f|_{x=0}[s] = p(s)$.

When we want to emphasize that the homogeneous polynomial arises from the derivative of a function we will use the notation $D^{\otimes k} f$, otherwise we use the notation p. Given this, we now state Taylor's Theorem. While the remainder could be stated to be composed of partial derivatives of the function, we define it only via being a bounded function of a given order as it suffices for later analysis.

Theorem 2 (Taylor's Theorem)**.** *Suppose $f : \mathbb{R}^d \to \mathbb{R}$ is a $k+1$ times continuously differentiable function on a compact set D and let $M = \sup\{|\partial^\alpha f(x)| : |\alpha| = k+1, x \in D\}$. If $x \in D$ and $x + s \in D$, then*

$$f(x+s) = \sum_{i=1}^{k} \frac{1}{k!} D^{\otimes k} f|_x[s] + R_{x,k}(s),$$

where $R_{x,k}(s)$ is called the remainder and satisfies $|R_{x,k}(s)| \leq \frac{M}{(k+1)!}(\sum_{i=1}^d |s_i|)^{k+1} =: R_{x,k}^{(ub)}(s)$.

2.2.3 Error Decomposition: Approximation and Noise Error Tradeoff

The centerpiece of local gradient estimation is the approximation and noise error tradeoff. To provide a first insight we look at the simple case of a directional derivative estimate via forward differences. Assume that the black box function has constant additive noise with constant variance. Specifically, it is given by $\tilde{f}(x; i) = f(x) + \epsilon(i)$ with $\epsilon \sim \mathcal{N}(0, \sigma^2)$. We look at the mean squared error of this estimate and use h as the step-size giving the following decomposition

$$\mathbb{E}[(\frac{\tilde{f}(x+vh) - \tilde{f}(x)}{h} - \partial_v f(x))^2] = \underbrace{\left(\frac{f(x+vh) - f(x)}{h} - \partial_v f(x)\right)^2}_{\text{Approximation Error}} + \underbrace{\frac{2\sigma^2}{h^2}}_{\text{Noise Error}}.$$

The decomposition is a result of the independence of noise and its distribution having zero mean and constant variance σ^2.

Now, using Taylor's Theorem on $f(x + vh)$ we have that for the approximation error

$$\frac{f(x + vh) - f(x)}{h} - \partial_v f(x) = \frac{1}{2}\partial_v^2 f(x)h + \mathcal{O}(h^2).$$

Notice that the first term depends directly on the second derivative of the function and scales as h. Higher-order errors are encapsulated in the term $\mathcal{O}(h^2)$.

Before discussing how to choose a good step size, h, we first observe the competing effects of the noise and approximation errors. For this we observe how the mean squared error depends on the step-size

$$\mathbb{E}[(\frac{\tilde{f}(x + vh) - \tilde{f}(x)}{h} - \partial_v f(x))^2] = \mathcal{O}(h^2) + \mathcal{O}(\frac{1}{h^2}).$$

To reduce the approximation error, we need to decrease h. However, this leads to an increase in the noise error. This is intuitive, as a smaller h will lead to more variance in the gradient approximator due to noise. To observe this, consider Figure 2.1. Here we plot regimes in which either the approximation error is dominating and where the noise error is dominating. When the approximation error dominates then the gradient estimator has reduced variance but higher bias. When the noise error dominates, the bias may be lower, but the variance is increased.

To reduce the error, more samples often help. Some methods focus on removing the lowest-order approximation error. For example, a central difference approach will remove the second-order error, however, it will require twice the number of samples. Similarly the noise error can be mitigated by simply sampling more points. Then by a standard central limit argument the standard deviation will scale as $\frac{1}{\sqrt{N}}$.

The other approach to reduce this error is to choose the best step size h. For this, if we knew the second derivative $\partial_v^2 f$, we could solve this problem exactly. Here, we would get an optimal $h = \left(2\sqrt{2}\frac{\sigma}{|\partial_v f(x)|}\right)^{1/2}$. This is simply given by taking the derivative of the truncated mean squared error, setting it to zero, and solving for h.

Here we clearly see the approximation and noise error trade-off in the size of h. The larger the

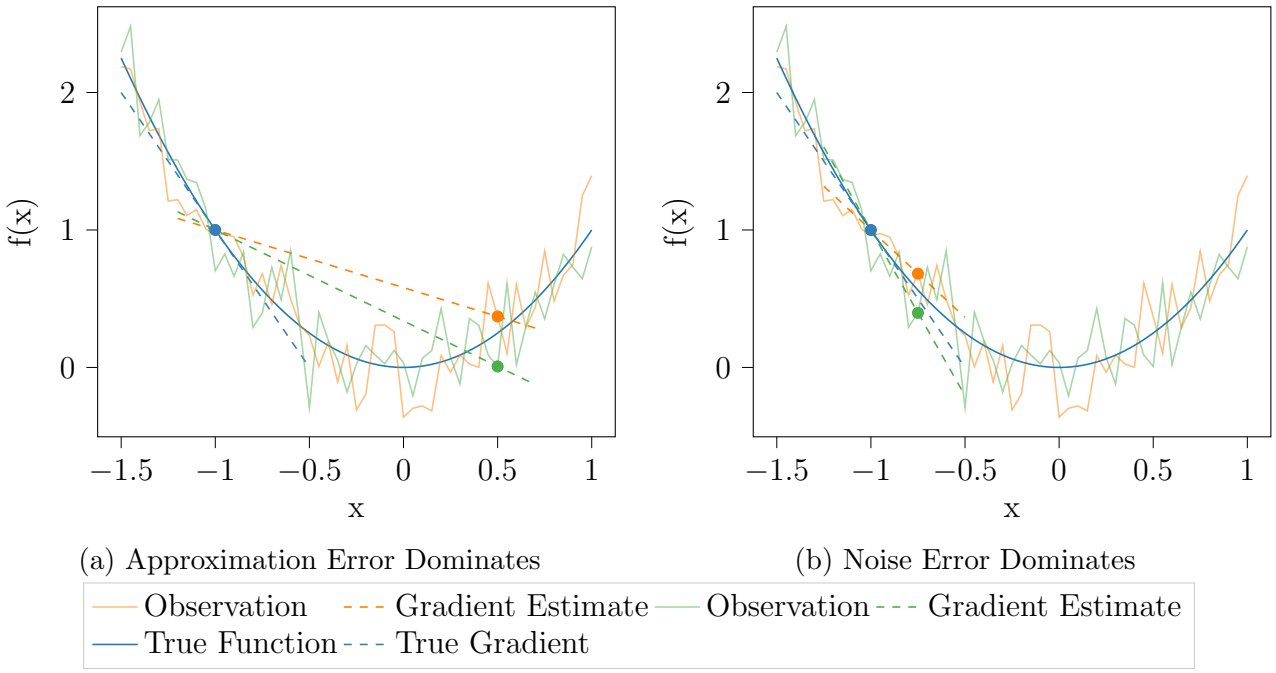


Figure 2.1: Visualizing the approximation and noise error tradeoff. Two samples of the black-box system are used to estimate the gradient at the given location via forward differences.

second-order component, the smaller the h should be to mitigate the increased approximation error. A similar argument holds for the noise.

Now, for a full gradient estimate, we are not required to sample in the direction of v. So, we may choose different sample points which may have better second order properties. Specifically, there may be some directions in which $\partial_v^2 f(x)$ is close to zero. These directions indicate regions in which the gradient does not change too much. Then, we can simply increase h in those directions to mitigate the noise error better.

The above observation will form the basis for the motivation of our two main results. In essence, we want to exploit the structure of the lowest-order errors to better mitigate the error without having to sample more points.

2.3 Gradient Estimation Models

We now discuss various existing gradient estimation methods. We distinguish between linear models and non-linear models. We put specific emphasis on linear models, as many gradient

estimation methods have an interpretation as a linear model. In the linear model discussion, we emphasise the role of the simplex gradient and its connection to other methods.

2.3.1 Linear Models

A linear model can often be thought of a model call which uses observations of the function to fit a function of the form

$$\mathcal{F}(y) = c + w^T y \tag{2.3.1}$$

to approximate the function locally at x. The gradient estimate at x is then given by w. To formalize this, the class of fully linear models can be defined [AH17].

Definition 3. *Given $f \in \mathcal{C}^1$, $x \in \mathbb{R}^d$, and $\bar{\Delta} > 0$ we say that $\{\mathcal{F}_\Delta\}_{\Delta \in (0,\bar{\Delta}]}$ is a class of fully linear models of f at x parameterized by Δ if there exists a pair of scalars $\kappa_f(x) > 0$ and $\kappa_g(x) > 0$ such that, given any $\Delta \in (0, \bar{\Delta}]$ the model $\mathcal{F}_\Delta$ satisfies*

$$|f(y) - \mathcal{F}_\Delta(y)| \leq \kappa_f(x)\Delta^2 \qquad \text{for all } y \in B_\Delta(x)$$
$$\|\nabla f(y) - \nabla \mathcal{G}_\Delta(y)\|_2 \leq \kappa_g(x)\Delta \qquad \text{for all } y \in B_\Delta(x)$$

where $\mathcal{G}(x) = \nabla \mathcal{F}(x)$.

The definition of the linear model becomes clearer when considering f in its Taylor approximation. Here we see that the fully linear model $\mathcal{F}$ ought to approximate the first-order term, that is, the gradient, of this expansion.

Using a linear model is often beneficial as it is simple to explain and is generally robust to overfitting. Furthermore, by being linear, it inherently approximates the underlying quantity of interest, the gradient, well. Higher-order methods will be prone to picking up errors due to noise. Due to the flexibility of higher-order models, they may attempt to explain the data by fitting noise, which generally hurts model performance.

In the following, we first discuss the estimator based on a least-squares linear fit, called the simplex gradient. We then discuss many related methods which have an interpretation as the simplex gradient and hence can be seen as an instance of it.

Simplex Gradient

Given a sample set, the gradient obtained by a best linear fit is called the *simplex gradient*. Given some data one minimizes the mean squared error of a linear model, such as in Equation 2.3.1, to fit the model. In our discussion, we focus on the case where the simplex gradient is obtained by a fully adaptive scheme, that is by first selecting a sample set $\mathcal{X}$, then querying the system $\tilde{f}$ and using the resulting data to obtain the best linear fit $\mathcal{F}$. We now provide the notation used.

The sample set is now given by an ordered set $\mathcal{X} = \{x_i\}_{0 \leq i \leq N}$ and by $N+1$ elements. Often x_0 is assumed to be the location of the gradient estimate, but this is not always a requirement. The **difference set** is given by $\mathcal{S} = \{x_i - x_0\}_{1 \leq i \leq N}$ and each element is called a **difference vector**. The jth difference vector is denoted denoted by s_j. The corresponding **difference matrix** is a matrix $S \in \mathbb{R}^{d \times N}$ which concatenates all difference vectors, i.e. $S_{ij} = (x_i - x_0)_j = (s_i)_j$. The **function vector** is given by $\tilde{\mathbf{f}}_{\mathcal{X}}$, where each component is set as $(\tilde{\mathbf{f}}_{\mathcal{X}})_i = \tilde{f}(x_i)$. The **function difference vector** is given by $\delta\tilde{\mathbf{f}}_{\mathcal{X}}$ and is given component-wise as $(\delta\tilde{\mathbf{f}}_{\mathcal{X}})_i = \tilde{f}(x_i) - \tilde{f}(x_0)$.

We now introduce the only necessary assumption on the sample set such that the simplex gradient exists. Then we provide the definition of the simplex gradient [AH17, BCCS22, BK98, CT19].

Assumption 2. *The difference set $\mathcal{S}$ spans the space $\mathbb{R}^d$. Or equivalently, $N \geq d$ and the difference matrix S has full rank.*

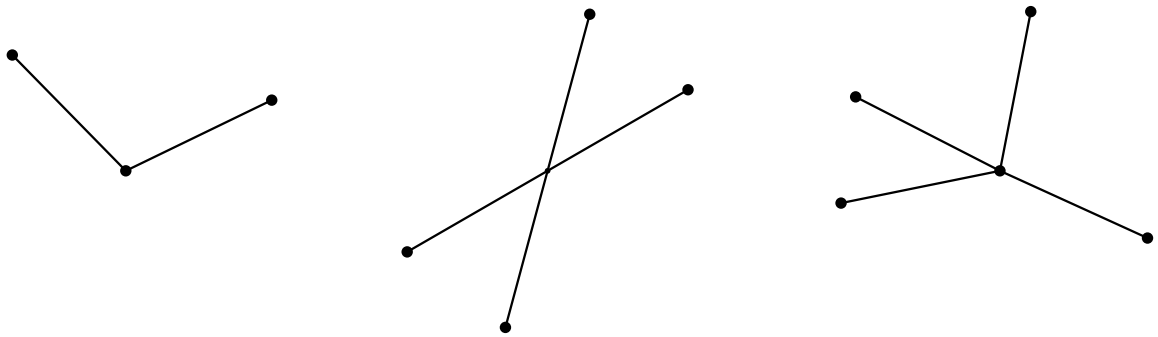
Definition 4. *Let $f : \mathbb{R}^d \to \mathbb{R}$ and let $\mathcal{X}$ satisfy Assumption 2. The simplex gradient of f over $\mathcal{X}$, denoted $\nabla_S f(\mathcal{X})$*

$$\nabla_S f(x_0) = S^+ \delta f(\mathcal{X}).$$

Given the model definition, the choice of the sample set is the central problem of the simplex gradient. As we have seen in the simple example of the approximation and noise error tradeoff in Section 2.2.3, the step-sizes can have a significant impact on the accuracy of the gradient estimate. As we will see in the following sections, many developed methods can readily be interpreted as simplex gradient, with the only difference between the methods being the choice of the sample set. Understanding what choice of the sample set is optimal or developing a framework to assess optimality is then crucial. With such results, we can then assess the performance of other methods and better understand their relationship to each other. Furthermore, such results then enable the development of further principled gradient estimation methods.

Two of the most prominent choices of the sample set, and the ones we will study in the thesis, are presented in Figure 2.2. The first sample set given in Figure 2.2a illustrates a fully-determined sample set. This sample set uses $d + 1$ sample points and included the point of gradient evaluation in the sample set in the first position. Given the use of $d + 1$ function evaluations, this linear model fully interpolates the data. While some methods have some understanding of the placement of the sample set, there is no principled methodology available yet which does not severely restrict the geometry [AH17, BCCS22, BFB15]. This will be the central study in Chapter 3.

The sample set in Figure 2.2b represented the centered simplex gradient. This sample set uses $2d$ sample points and does not have the point of evaluation in the sample set. It is characterized by the symmetry in the sample point selection. This symmetry allows it to obtain significantly improved error qualities. Once again, there does not exist a principled sample set selection result beyond either methods which are heuristically derived or restrict the geometry [HJBP22, CH23, BFB15]. This will be the central study in Chapter 4.



(a) Fully Determined Simplex Gradient (b) Centered Simplex Gradient (c) Simplex Gradient

Figure 2.2: The structure of various simplex gradient instances and formulations. Dots representing sample points.

During the discussion of the remaining gradient estimation methods, we pay specific attention to their relation to the simplex gradient. By seeing that many developed gradient estimation methods are either direct instances of the simplex gradient or are similar, we further emphasise the importance of principled sample set selection. Specifically, due to the difference between simplex gradient based methods lying in the choice of the sample set.

Finite Difference Schemes

Finite differences schemes are at the centre of numerical differentiation schemes. These mostly focus on estimating a directional derivative in each of the coordinate directions and concatenating them into a single gradient estimate. Besides defining the methods, we will also provide a heuristic argument for the relationship between these methods to the simplex gradient.

To estimate the directional derivative, finite difference methods choose a set of step-sizes $\{h_i\}_{1\leq i\leq N}$ and a set of weights $\{w_i\}_{1\leq i\leq N}$ [BFB15, BH12, Cul71]. They then use these to estimate the gradient in the e_k coordinate direction as follows

$$\partial_{e_k} f(x) \approx \sum_{i=1}^{N} w_i \tilde{f}(x + h_i e_k).$$

Instead of implicitly concatenating the estimates, we can simply choose a set of weight vectors

$\{w_i \in \mathbb{R}^d\}_{1 \leq i \leq N}$ and a sample set $\mathcal{X} = \{x_i\}_{1 \leq i \leq N}$ to define the estimate via

$$\nabla f(x) \approx \sum_{i=1}^{N} w_i f(x_i). \tag{2.3.2}$$

For the canonical examples the generalization via the simplex gradient is immediate. Consider forward differences given by the weights $\frac{1}{h}$ and the step-sizes are given by $\{0, h\}$. This can immediately be achieved by a fully-determined simplex gradient with a sample set given by $\{x\} \cup \{h_k e_k\}_{1 \leq i \leq d}$, with e_k being the step-sizes in each coordinate direction. Central differences is given by weights $\frac{1}{2h}$ and step-sizes $\{-h, h\}$. Here the centered simplex gradient uses the sample set $\{-h_k e_k\}_{1 \leq i \leq d} \cup \{h_k e_k\}_{1 \leq i \leq d}$. Both methods, severally restrict the geometry of the sample set by assuming the sample points to be along the coordinate directions. It remains an open question what a well-chosen sample set without such restrictions may be.

There are various variations to derive these weights and sample sets. One of which is given by Richardson Extrapolation, which uses increasingly more sample points to improve error properties [BFB15, BH12, KR14, BDHdH08, For88].

We now heuristically argue that the chosen weights and step-sizes naturally imply the use of a simplex gradient. We will do so by showing that for the model to be a fully linear model as given in Definition 3, the weights have to stem from an inverse of the difference matrix of the sample set. For this argument, we assume to take weighted average of the differences of $f(x_i) - f(x)$ [1]

$$\nabla f(x) \approx \sum_{i=1}^{N} w_i (f(x_i) - f(x)).$$

Now, let W be the concatenation of all the weights and consider the first order Taylor expansion

[1] This removes the zeroth order term in the Taylor expansion. When W is zero mean, meaning that $\sum_{i=1}^{N} w_i = 0$, then we do not need to subtract $f(x)$ as the zeroth order term is naturally removed.

of $f(x)$ yielding

$$\sum_{i=1}^{N} w_i(f(x+s_i) - f(x)) = \sum_{i=1}^{N} w_i \left((\nabla f(x))^T s_i + \frac{1}{2} D^2 f|_x[\xi_i] \right)$$
$$= WS^T \nabla f(x) + R_f(\mathcal{X})$$

where for ξ_i there exists $\lambda_i \in (0, 1]$ such that $\xi = \lambda_i x + (1 - \lambda_i)x_i$ and $R_f(\mathcal{X})$ is the accumulation of the second order error terms. Also, recall that S provides the difference matrix which we defined along with the simplex gradient in Section 2.3.1.

To make the method a fully linear model, according to the above definition, we need a relation in which reducing a parameter Δ leads to a more accurate estimate. In this case, we let Δ be the radius of the sample set, given by $R(\mathcal{S})$. Recall that $R(\mathcal{S})$ is the length of the largest difference vector. For the error to decrease as the radius is the decreased, we need the first term, given by $WS^T \nabla f(x)$, to be exactly equal to the gradient. Otherwise, the systematic error in the gradient estimate will remain even as the radius is reduced.

For the first-order term to equal the gradient, we need $WS^T = I$. If the number of difference vectors is given by d, then $W = S^{-T}$. When there are more sample points, there is no unique solution. When using the minimum norm solution, then W is simply given by the pseudo-inverse of S^T. Hence, the form now becomes equivalent to the simplex gradient.

Likelihood Ratios and Monte Carlo Estimates

The likelihood ratio provides a range of stochastic gradient estimation methods. It is based on a simple integration by parts argument which provides the relation

$$\mathbb{E}[\nabla f(Y)] = \mathbb{E}[f(Y) \nabla \log p(Y; x)]$$

where $p(\cdot; x)$ is a distribution depending on the gradient estimation location x [Gly90, TS17]. It can then be shown that as either the support shrinks around x, or the variance decreases, that the estimator approaches the exact gradient in $\nabla f(x)$. These methods also often generalize

general stochastic approximation methods [RK16, RB16, Bor22] or smoothing techniques, as the right-hand side can be viewed as the gradient of a function smoothed against the density p [BCCS19, Wu96, KO72].

We now want to emphasise the relationship to the above numerical differentiation schemes and hence the simplex gradient. For this, note that the Monte-Carlo estimates which are often used to approximate such expectations, will have a form reminiscent of the numerical differentiation form given in Equation 2.3.2. This is because a finite number of samples from the distribution will be taken to then evaluate the function and compute the weights. The performance of these methods will then depend on the choice of the distribution. Hence, the form is not guaranteed to be exactly as the one of the simplex gradient and of the gradient of a least squares fit. As we established in the previous section on numerical differentiation, W would have to be a pseudo-inverse of the sample set produced by sampling from the distribution p.

A commonly used density for which the relation to the simplex gradient is immediate is for a normal distribution. Expanding the likelihood ratio we obtain $\mathbb{E}[f(Y)\nabla \log p(Y;x)] = \Sigma^{-1}\mathbb{E}[f(X)(X-x_0)]$. This is also the result of Stein's lemma [DJR13] and of using a smoothing approach [BBN19].

To see the direct relationship with the simplex gradient, let $\mathcal{X}$ be a sample set which has the same first and second moment as the given normal distribution. Letting S be the corresponding difference matrix, we can use this as an estimator of the expectation, yielding $\Sigma^{-1}S\delta f(\mathcal{S})$. Noting that Σ^{-1} is also equal to the sample variance by construction of $\mathcal{X}$, we obtain that the estimator is exactly the simplex gradient $S^{+}\delta f(\mathcal{S})$.

When using a true Monte Carlo estimator, the sample set will not exactly match the moments of the true distribution. So, there will be some variance due to sample set deviation. We may ask whether this can be helpful. It turns out that the gradient estimator does not benefit from this. In many cases, the variance due to sampling the sample set from some distribution increases the gradient estimation error [BCCS19, BBN19, BCCS22, Sch22]. Of course, in some situations the induced variance may be helpful. We have in fact found that the variance can bias an optimization method in some beneficial ways. So while it may not be as good of a gradient

estimator, the induced distribution may be beneficial.

Simultaneous Perturbation Stochastic Approximation and Related Methods

In some applications, only very few function evaluations can be performed. In such a case, the sample set describes an under-determined system, for which a simplex gradient formulation exists [CSV08a, CSV08b, HJBP22]. While other methods exist, the most famous method is simultaneous perturbation stochastic approximation (SPSA) which uses only two function evaluations, regardless of the dimensionality of the parameter space [S+92, Spa98, BPP13, BDS17, FKM04].

Instead of perturbing each parameter individually, SPSA perturbs all parameters simultaneously in a random direction. This is done by generating a random perturbation vector χ, where each component is typically chosen from a symmetric distribution (e.g., Bernoulli distribution). The gradient is estimated using the difference in function values at perturbed points $\nabla f(x) \approx \frac{f(x+c\chi)-f(x-c\chi)}{2c} \cdot \chi^{-1}$. Here, c is a small positive constant, and χ^{-1} represents the element-wise inverse of the perturbation vector.

We can see this is a two element Monte Carlo estimate of a likelihood estimator. In expectation, it can also be shown to be a consistent estimator, that is, in expectation the Monte Carlo estimates equal the true gradient. However, such methods are often far too inaccurate for gradient estimation [BZ19] and have been shown to have poorer performance compared to a deterministic counterpart [Sch22, BCCS19, BCCS22].

Design Of Experiments

While previous methods were restrictive in their geometry (numerical differentiation) or stochastic (likelihood ratio and SPSA) Design of Experiments (DoE) methods place sample points freely and deterministically [DB03, BFB15, ZS91, BDHH05]. However, they do not have further optimality conditions associated with them and are heuristically motivated.

DoE was pioneered in assessing the factors that influence the response of a system by establishing principles by which to assess the individual and combined impact of changing the parameters of the system in an efficient way [Fis36]. These methods have resulted in various schemes, such as the fractional factorial design [Mon17] and the Plackett–Burman design [PB46]. In addition to the absence of formal guarantees of optimality, their performance is generally comparable to or worse than standard finite difference schemes, making them so far the preferable option [BDHH05].

2.3.2 Non-linear models

Non-linear models are significantly more expressive than linear models and hence can capture higher-order behaviour which may make the fitted function more accurate. The hope is that when the right model is chosen, that points can be placed more freely while retaining a good approximation of the function. An example of this is knowing that the true function may be a uni-variate cubic. Then, as long as four distinct function evaluations are available, the function can fit exactly. In this way, we would not require local function evaluations as the function can be fit with any set of data.

There are two significant difficulties associated with the use of non-linear models. For one, there has to be a strong understanding of the underlying function and its structure. Furthermore, the effect of noise may remain to be difficult to mitigate. In the case of very expressive models, one will often observe a strong overfitting phenomenon, where the model will explain the noise rather than the true behavior or numerical instability [CL09, DB03].

To mitigate the effect of noise, not only will the function class have to be understood, but significant regularization will be required. This type of regularization may be in terms of the size of the parameters used in the model or a bound on function dependent quantities such as the Lipschitz constant [HS01, AA07, Cul71, KR14, WH07, VBKB20, BDHdH08]. Hence, a linear model is often preferable, as it is guaranteed to locally be a strong approximator of any sufficiently smooth function.

2.4 Applications of the Simplex Gradient

Gradient estimation is prevalent in multiple domains. We now review two of the most impactful ones, being sensitivity analysis and derivative free optimization. For these methods, the core focus remains in having to select appropriate sample sets for a gradient estimate. These methods also often operate in local and adaptive settings, further justifying our focus in this thesis.

In our study of the simplex gradient we found a prevalence of implicit gradient estimation in many unexpected fields. This was also the initial motivation for studying gradient estimation methods, as we were aiming to better understand seemingly tangential methods. We will briefly outline these more heuristic connections in the conclusions in Chapter 5.

2.4.1 Sensitivity Analysis

The goal of sensitivity analysis is to understand the impact of changes in the output of a system given changes to the input [VA00]. A common application is in biology and chemistry to understand the effect of parameters on treatments and biological processes [QM20, JEB+07, DPV06]. Other applications involve financial and economic models to study system behaviors and critical inputs [GLS00, Pan97].

In many cases, where the function to be studied is differentiable, sensitivity analysis considers the estimation of gradients with respect to some parameter of interest [VA00, KIT+19]. Here, one is interested in understanding to first order the impact of the parameters on the output. The gradient is then often estimated via a best linear fit estimate via regression analysis [TMZ+13, Sob09, IL15]. The best linear fit in our notation is simply given by the simplex gradient, making it a key tool for sensitivity analysis.

2.4.2 Derivative Free Optimization and Meta Heuristic Algorithms

Another core application of the simplex gradient is in derivative free optimization (DFO) and meta-heuristic algorithms. Here, the goal is to minimize some objective while only having access

to evaluations of a black-box system [LMW19, SBK+20, HRR23, ABAFS18, BKW24, Bor22]. Such methods have wide ranging applications from finance [GMS19, CT06], chemical product design [SSSC20], optimization of cardiovascular geometries [MFT08] to trajectory control and drone swarms [LK17, MCDDK16, KBD04]. Common algorithms include trust region methods and local surrogate models [MWDS18, SBK+20, CHJB24], pattern search methods [CV07, CDV08] and other famous method such as the Nelder-Mead algorithm [NM65]. Some of these methods can also be found in the statistic literature, where a more clear connection to DFO methods is currently being pursued. Two examples are iterated filtering, where the update step follows a likelihood ratio update [IBAK11], and the inverse Kalman filter via statistical linearization [CCSA20].

Naturally, a key aspect of many DFO aspects is finding a descent direction. For many algorithms, the gradient estimation is explicitly defined and studied [BCCS22, Spa98, Bor22, CCI16, FCJR17, SQXON23, Kel11]. In other cases, gradient estimation is not explicitly defined and only later found to be a core aspect of the algorithm [BK98, CCSA20, DJR13]. Therefore, for many problems, the core expense involves function observation as part of gradient estimation. By better understanding optimal sample set structures, we can then hope for the development of improved derivative free optimization methods. These can then have a large impact on the aforementioned application domains.

2.5 Summary

In this chapter, we established the gradient estimation problem and laid out our assumption of studying the problem in a local and adaptive setting. In this context, we have made all assumptions on function and noise structure explicit and assessed their practical viability. We then discussed various gradient estimation methods, contrasting linear and non-linear models. In the linear setting, we argued that the simplex gradient is a generalization of many commonly used gradient estimation methods. For also discussed the robustness and stability of linear methods when little about the function structure is known. When the function structure is

understood better, then non-linear methods may be beneficial. We concluded the chapter with a summary of applications and the presence of the simplex gradient in these. This presence of the simplex gradient highlights the importance of studying the optimal sample set structures, as it leads to cost savings in a range of applications by reducing the number of function evaluations while retaining or improving on accuracy. The following chapters will focus on the most commons setting, the fully-determined and centered simplex gradient.

Chapter 3

Optimal Sample Sets Selection for the Fully-Determined Simplex Gradient

In this chapter we study the optimal sample set of the fully-determined simplex gradient. Recall from Section 2.3.1, that the fully-determined simplex gradient is a best linear approximation of the function locally and uses $d+1$ sample points. Hence, it requires the fewest number of function evaluations of any valid simplex gradient. This makes it an important subject to study, as it allows us to understand the properties of gradient estimators which try to reduce cost as much as possible.

To accomplish the above, we proceed as follows. We recall the background and general problem statement established in Chapter 2. Due to using the fully-determined simplex gradient, the expression for the simplex gradient simplifies to $S^{-T}\delta\tilde{f}(\mathcal{X})$. In Section 3.1 we motivate and derive a Taylor-expansion based objective function for the optimization problem related to finding the optimal sampling set for the simplex gradient and derive some crucial simplifications. In Section 3.2 we present the main results and provide the proof and algorithms for CASG. We additionally provide some intuition behind CASG and an initial comparison to forward differences. In Section 3.4.3 we then present the extended curvature aligned simplex gradient. We provide the global model framework algorithm and the empirical results in Section 3.5. In Section 3.5 we also discuss in further detail the use of the global model as a separate gradient

estimation method. We also consider the cost of the global model framework when no history of function evaluation is available.

3.1 Optimization Problem for Optimal Sample Set

In this section we motivate and define the objective function which is used to assess the goodness of a sample set for numerical differentiation under the simplex gradient. The objective function is based on the mean squared error and is derived via a Taylor-expansion argument as is common for finite difference schemes [BFB15].

To better understand the mean squared error, we consider its decomposition into the approximation and noise error via the following Proposition.

Proposition 5 (MSE Decomposition)**.** *Let the sample set $\mathcal{X}$ and the black-box function $\tilde{f}$ be given and given Assumption 1 on the noise. Let S be the difference set corresponding to the sample set. Then the mean squared error is given by*

$$\begin{aligned}&\mathbb{E}[\|\nabla_S \tilde{f}(x_0) - \nabla f(x_0)\|_2^2] \\ &= \underbrace{\|\nabla_S f(x_0) - \nabla f(x_0)\|_2^2}_{\textit{Approximation Error}} + \underbrace{\sigma^2 \|S^{-1}\|_F^2 + \sigma^2 \|S^{-T}\mathbf{1}\|_2^2}_{\textit{Noise Error}}.\end{aligned}$$

Proof. We expand the mean squared error after which we proceed term by term

$$\begin{aligned}\mathbb{E}[\|\nabla_S \tilde{f}(x_0) - \nabla f(x_0)\|_2^2] &= \mathbb{E}[\|\nabla_S \tilde{f}(x_0) - \nabla_S f(x_0) + \nabla_S f(x_0) - \nabla f(x_0)\|_2^2] \\ &= \mathbb{E}[\|\nabla_S \tilde{f}(x_0) - \nabla_S f(x_0)\|_2^2] + \|\nabla_S f(x_0) - \nabla f(x_0)\|_2^2 \\ &\quad + 2\mathbb{E}[\Big(\nabla_S \tilde{f}(x_0) - \nabla_S f(x_0)\Big)^T \Big(\nabla_S f(x_0) - \nabla f(x_0)\Big)].\end{aligned}$$

By linearity of the simplex gradient we have that $\mathbb{E}[\|\nabla_S \tilde{f}(x_0) - \nabla_S f(x_0)\|_2^2] = \mathbb{E}[\|\nabla_S \epsilon(0)\|_2^2]$ where we write $\nabla_S \epsilon(0) = S^{-T}\delta\epsilon(\mathcal{X})$ with $\delta\epsilon(\mathcal{X}) = [\epsilon(1) - \epsilon(0), \ldots, \epsilon(d) - \epsilon(0)]^T$. We then have

for the first term

$$\begin{aligned}\mathbb{E}[\|\nabla_S \epsilon\|_2^2] &= \mathbb{E}[\delta\epsilon(\mathcal{X})^T S^{-1} S^{-T} \delta\epsilon(\mathcal{X})] \\ &= \operatorname{tr}\left(S^{-1}S^{-T}(\sigma^2 I + \sigma^2 \mathbf{1}\mathbf{1}^T)\right) \\ &= \sigma^2 \|S^{-T}\|_F^2 + \sigma^2 \|S^{-T}\mathbf{1}\|_2^2,\end{aligned}$$

where we used the result from [Ken05] that for any random vector $\mathbf{X}$, with mean μ and covariance matrix Λ, and symmetric matrix A we have $\mathbb{E}[\mathbf{X}^T A \mathbf{X}] = \mu^T A \mu + \operatorname{tr}(A\Lambda)$. The second term $\|\nabla_S f(x_0) - \nabla f(x_0)\|_2^2$ is already in the correct form. Lastly, the cross term $\mathbb{E}[\Big(\nabla_S \tilde{f}(x_0) - \nabla_S f(x_0)\Big)^T \Big(\nabla_S f(x_0) - \nabla f(x_0)\Big)]$ is easily seen to equal zero since $\mathbb{E}[\nabla_S \tilde{f}(x_0)] = \nabla_S f(x_0)$. □

While the noise error only depends on the noise level and distribution of points, the approximation error also depends on the function structure. Since we do not have access to the true function f, we need use some approximation of f. However, even with an approximation of f, the MSE may be difficult to minimize directly. Instead, we introduce a bound on the approximation error.

3.1.1 Approximation Error Bound

While there exist multiple methods by which to estimate the approximation error, we proceed via a Taylor argument as done in many other works [HJBP22, BFB15, BH12]. The crucial difference is that previous works upper bound the second order term, while we retain the second order term exactly and only bound the higher order terms.

Lemma 6. *Let $x_0 \in \mathbf{R}^d$, $f \in \mathcal{C}^2$, $H = \nabla^2 f(x_0)$ and $\|S\|_2 \leq h$ for some $h \in \mathbb{R}_+$. Then we have*

$$\|\nabla_S f(x_0) - \nabla f(x_0)\|_2^2 = \frac{1}{4}\|S^{-T} p_H(\mathcal{S})\|_2^2 + \mathcal{O}(h^3).$$

Proof. By Taylor's Theorem for a fixed x_0 and $x \in \mathcal{X}$ we have

$$f(x) - f(x_0) = (\nabla f(x_0))^T s_i + \frac{1}{2} s_i^T H s_i + \mathcal{O}(\|S\|_2^3)$$
$$\rightarrow \nabla_S f(x_0) = S^{-T}\delta\tilde{\mathbf{f}} = \nabla f(x_0) + \frac{1}{2} S^{-T}\text{diag}(S^T H S) + \mathcal{O}(\|S\|_2^2).$$

We then have the following bound on the difference of the simplex gradient and the true gradient

$$\begin{aligned}\|\nabla_S f(x_0) - \nabla f(x_0)\|_2^2 &= \|S^{-T}\delta\tilde{\mathbf{f}} - \nabla f(x_0)\|_2^2 \\ &= \|\frac{1}{2} S^{-T}\text{diag}(S^T H S) + \mathcal{O}(\|S\|_2^2)\|_2^2 \\ &= \frac{1}{4}\|S^{-T}\text{diag}(S^T H S)\|_2^2 + \mathcal{O}(\|S\|_2^3)\end{aligned}$$

where the last step follows from the triangle inequality and noting that $\frac{1}{4}\|S^{-T}\text{diag}(S^T H S)\|_2^2 = \mathcal{O}(\|S\|_2)$. Since the norm of S is bounded by h the result follows.

□

3.1.2 Objective Function

When using the bound on the approximation error and the noise error, the only variables are the noise level σ, the given Hessian matrix H, and the sample difference set S. To control the error due to higher-order effects, the hyperparameter h is introduced. The optimization problem is then given by the following.

Definition 7. *Let $H \in \mathbb{R}^{d\times d}$ be symmetric, $\sigma \in \mathbb{R}_+$ and $h \in \mathbb{R}_+$. Let $\boldsymbol{AE} = \frac{1}{4}\|S^{-T} p_H(\mathcal{S})\|_2^2$ be the approximation error and $\boldsymbol{NE} = \sigma^2\|S^{-1}\|_2^2 + \sigma^2\|S^{-T}\mathbf{1}\|_2^2$ be the noise error. We then define the function $\ell_{H,\sigma,h} : \mathbb{R}^{d\times d} \rightarrow [0, \infty]$ as*

$$\ell_{H,\sigma,h}(S) = \begin{cases} \boldsymbol{AE} + \boldsymbol{NE} & \text{if } h \geq \|S\|_2 \text{ and } S \in GL_d \\ \infty & \text{, else.} \end{cases}$$

When clear from context, we drop the H, σ, h subscript and write $l(S)$. Furthermore, we let Ω

be the subset in which the objective function is finite.

Remark 1. *The objective function is well defined. The function is continuous on the compact domain $\bar{\Omega}$ given by the intersection of $h \geq \|S\|_2$ and the closure of GL_d under the topology for the extended real number line. Hence, it attains its minimum in $\bar{\Omega}$. Note that without an upper-bound on the size of the difference matrix, the objective function may not be well-defined.*

The following Proposition significantly simplifies the objective function and allows us solve the optimization problem in the basis in which H is diagonal.

Proposition 8. *Let H be a symmetric matrix and let R be an orthogonal matrix such that $R^T H R = D$. Defining $S' = R^T S$ then $\ell_{H,\sigma}(S) = \ell_{D,\sigma}(S')$. Furthermore, $\ell_{D,\sigma}(S) = \ell_{-D,\sigma}(S)$.*

Proof. Since H is symmetric such an orthogonal matrix R exists. Then $\|S^{-1}\|_F = \mathrm{tr}(R^T S'^{-T} S'^{-1} R)$ and by the cyclic property of trace $\|S^{-1}\|_F = \|S'^{-1}\|_F$. Furthermore, since orthogonal operations do not change norms $\|S^{-T}\mathbf{1}\|_2^2 = \|S'^{-T}\mathbf{1}\|_2^2$. For the approximation error note that $s_i^T H s_i = s_i'^T D s_i'$. As established, orthogonal matrices do not change norms and hence $\|S^{-T} p_D(\mathcal{S})\|_2^2 = \|S'^{-T} p_D(\mathcal{S}')\|_2^2$. The invariance to negating the diagonal is due to the absolute homogeneity property of the norm. □

While more manageable, the objective function is not trivial to minimize due its remaining non-convexity and non-linearity. This is unlike standard forward difference where the difference matrix has orthogonal columns which allows the optimization problem to separate into a sum of one-dimensional convex functions which are easy to solve.

3.2 Theoretical Results

In this section we discuss the solutions to the sample set optimization problem introduced in the previous section via Definition **??**. We focus on finding the optimal solution $S^* = \arg\min_{S \in \mathbb{R}^{d \times d}} \ell_{H,\sigma,h}(S)$ for a H, σ and h. For this theoretical discussion, we assume these to be given values. That is, H is some symmetric matrix and $\sigma, h > 0$. Based on the theoretical

results, we then develop a methodology in Section 3.4 and numerically evaluate it in Section 3.5. The benefit of this theoretical discussion is understanding what sample set structure is beneficial for gradient estimation. The specific properties and intuition of this theoretically optimal sample set will be further discussed in Section 3.3.

The optimal sample set structure is derived based on the singular value decomposition (SVD) of the sample set. The SVD of the sample set is given by $U\Lambda V^*$ where U, V are orthogonal matrices and Λ is diagonal with the ith entry given by $\Lambda_i = \sqrt{\lambda_i}$. We then have that λ represents the square of the diagonal of Λ in vector form. The first result relates to the singular-vectors of the sample set. The choice of the singular-vectors is simple, yet allows for a significant simplification of the loss function. Even without deriving the optimal singular values, this simplification of the loss can lead to immediate improvements compared to standard methods as we will show in Section 3.3.

Theorem 9. *Assume that $\sigma, h > 0$, $H \in \mathbb{R}^{d\times d}$ is a symmetric matrix with the eigendecomposition $H = RDR^T$ and that there exists a d-dimensional Hadamard matrix. The global minimum of $\ell_{H,\sigma,h}$, S^*, is given by $RU^*\Lambda^*(V^*)^T$ where Λ^* is the solution to the following d-dimensional optimization problem*

$$\Lambda^* = \arg\min_{\Lambda} \begin{cases} \frac{(\sum_{i=1}^d D_i\lambda_i)^2}{4d\lambda_{max}} + \sigma^2 \sum_{i=0}^d \frac{1}{\lambda_i} + \sigma^2 \frac{1}{\lambda_{max}} d & \text{if } hI \succeq \Lambda \succ 0 \\ \infty & \text{otherwise} \end{cases}$$

and

- $U^* = I$
- $V^* = M_k$ *where* $k = \arg\max_k \Lambda_k^*$. *Recall that M_k is the Hadamard matrix with the entries of the kth row being $1/\sqrt{d}$.*

Note, the result relates to dimensions for which the Hadamard matrix exists. The current conjecture states that this exists for all dimensions which are multiples of four. When developing a method in Section 3.4 we address this limitation in more detail. We will also see in the next result that $V^* = M_1$, as the first singular value is the largest.

Given the above Theorem, we have transformed the optimization problem in Definition 7 to be more interpretable and reduced the number of variables from d^2 to d. The following theorem provides a solution to this optimization problem. While the closed form solution is complex, there are three basic properties providing an intuition for how the optimal singular values are structured. Furthermore, as we discuss in Section 3.3, this result allows us to study bounds on the loss function when using an optimal sample set. These results will provide intuition for how the choice of the singular values lead to an improvement over standard forward differences, and how other methods could use a similar approach to structure sample sets.

Theorem 10. *Let D be a diagonal matrix with increasing entries and K be the number of non-positive elements in D. Let $\mathcal{M}^*$ be the solution set to the optimization problem as given in Theorem 9. Then there exists a solution $\lambda^* \in \mathcal{M}^*$ with the following properties*

- *λ^* is in decreasing order,*
- *there exists $K \leq J \leq d$ such that $\lambda_i^* = h^2$ for $1 \leq i \leq J$ and $\lambda_i^* < h^2$ for $J+1 \leq i \leq d$,*
- *if $D_i = D_j$ then $\lambda_i^* = \lambda_j^*$ for $2 \leq i, j \leq d$.*

The solution λ^* is given by the following

$$\lambda_1^* = \begin{cases} h^2 & \text{if } 1 \leq J, \\ \frac{2d}{aD_1}\left(\frac{a^2}{4d} + \sigma^2(d+1)\right) & \text{otherwise;} \end{cases}$$

$$\lambda_i^* = \begin{cases} h^2 & \text{if } i \leq J, \\ \sigma\sqrt{\frac{2d\lambda_1}{aD_j}} & \text{otherwise,} \end{cases} \qquad \text{for } 2 \leq i \leq d,$$

with a given by

- *if $J = 0$*

$$a = \sqrt{2}\left(\frac{d\sigma^2}{D_1}\left(C\sqrt{8D_1(d+1)+C^2} + 2D_1(d+1) + C^2\right)\right)^{1/2}$$

with $C = \sum_{i=2}^{d} \sqrt{D_i}$;

- *otherwise*

$$a = \begin{cases} \frac{\sqrt[3]{\frac{2}{3}}c_2}{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}} + \frac{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}}{\sqrt[3]{18}} & \text{if } 27c_1^2 - 4c_2^3 \geq 0, \\ \cos(\theta/3)2\sqrt{\frac{c_2}{3}} & \text{otherwise,} \end{cases}$$

where $c_1 = \sigma\sqrt{\frac{dh^2}{2}} \sum_{i=J+1}^{d} \sqrt{D_i}$, $c_2 = \sum_{i=1}^{J} D_i h^2$ *and* $\theta = arccos(\frac{9c_1}{\sqrt{12c_2^3}})$.

Every other solution is a simple permutation of λ^*. *Specifically, for every* $\lambda \in \mathcal{M}^*$ *there exists* $j \in \arg\max_{1 \leq i \leq d} D_j$ *such that* $\lambda_1^* = \lambda_j$ *and* $\lambda_j^* = \lambda_1$ *and* $\lambda_k^* = \lambda_k$ *when* $k \neq j$ *for* $2 \leq k \leq d$.

3.2.1 Proof Outline of Theorem 9

To proof Theorem 9 we present three core propositions. We present the proof of each proposition in the Appendix 3.6.1. To simplify the proof, we make the following assumptions without any loss of generality. By Proposition 8, it suffices to left multiply a minimizer of $\ell_{D,\sigma,h}$ by R to obtain a minimizer for $\ell_{H,\sigma,h}$—we can then work with a diagonal matrix D rather than the symmetric matrix H. By the same proposition we know that the objective is invariant under negation of H, allowing us to assume that the trace of H and hence D is positive. Lastly, since the eigendecomposition of H is invariant under permutations of the eigenvalues in D, we assume without loss of generality that the entries in the diagonal of D are in increasing order.

First we derive the lower bound of the objective function ℓ. Notice that there is no more dependence on V, and hence this lower-bound holds for all possible V.

Proposition 11. *For a diagonal matrix* $D \in \mathbb{R}^{d \times d}$ *and positive real numbers* $\sigma, h \in \mathbb{R}_+$ *define the function*

$$\underline{\ell}_{D,\sigma,h}(U,\Lambda) = \begin{cases} \frac{a_{D,\sigma}(U,\Lambda)^2}{4d\lambda_{max}} + \sigma^2 \sum_{i=1}^{d} \frac{1}{\lambda_i} + \sigma^2 \frac{d}{\lambda_{max}} & \text{if } hI \succeq \Lambda \succ 0 \\ \infty & \text{otherwise,} \end{cases} \tag{3.2.1}$$

with orthogonal $U \in \mathbb{R}^{d\times d}$*, diagonal* $\Lambda \in \mathbb{R}^{d\times d}$*, and* $a_{D,\sigma}(U,\Lambda) = tr(\Lambda U^T D U \Lambda)$*. Then we have*

$$\underline{\ell}_{D,\sigma,h}(U,\Lambda) \leq \min_V \ell_{D,\sigma,h}(U,\Lambda,V).$$

We now present the optimal solution to the lower bound derived in the previous proposition. The optimal value for U is given by the identity matrix, while the optimal Λ is the solution to a simplified optimization problem and is the focus of Theorem 10.

Proposition 12. *The global minimizers* $\underline{U}^*$ *and* $\underline{\Lambda}^*$ *of* $\underline{\ell}_{D,\sigma,h}(U,\Lambda)$ *are given by* $\underline{U}^*$ *being the identity matrix and* $\underline{\Lambda}^*$ *being a diagonal matrix given by* $\min_\Lambda \underline{\ell}_{D,\sigma,h}(I,\Lambda)$*.*

Remark 2. *For an intuitive explanation for the above result, observe that* $a_{D,\sigma}(U,\Lambda) = \sum_{i=1}^d D_i(\sum_{j=1}^d \lambda_j U_{i,j}^2)$*. Since* U *is orthogonal,* $\sum_{j=1}^d \lambda_j U_{i,j}^2$ *is a weighted average of* λ_j*. Letting* U *be the identity matrix allows for the greatest degree of freedom, as each* λ_i *can address the corresponding* D_i*. The most limiting structure would be if* U *was the Hadamard matrix leading* $\sum_{j=1}^d \lambda_j U_{i,j}^2 = \frac{1}{d}\sum_{j=1}^d \lambda_j$ *and leaving only one degree of freedom.*

We now show that when V is the Hadamard matrix, the original objective function and the lower bound can be made equal. Specifically, they are equal when U is the identity and Λ is any valid diagonal matrix.

Proposition 13. *Let* $\Lambda \succ 0$ *be diagonal and* M_k *be the Hadamard matrix with* $k = \arg\max_i \Lambda_i$*. We then have* $\ell_{D,\sigma,h}(I,\Lambda) = \ell_{D,\sigma,h}(I,\Lambda,M_k)$*.*

Remark 3. *To achieve the lower-bound in Proposition 11 we needed* $\mathbf{1}$ *to be associated with the left-singular vector corresponding to the largest singular value of* S*. Furthermore, the approximation error in* $\|S^{-1}p(\mathcal{S})\|^2 \geq \frac{1}{\lambda_{\max}}\|p(\mathcal{S})\|^2$*. Due to a special property of* $p(\mathcal{S})$*, the sum of the elements is invariant over any orthogonal* V *and hence* $\|p(\mathcal{S})\|^2$ *is minimized when* $p(\mathcal{S}) = a_{D,\sigma}(U,\Lambda)\mathbf{1}$*. Incidentally, the matrix* V *which achieves both of these properties when using* U^* *is the Hadamard matrix.*

Given the three core propositions, we can now prove the first Theorem.

9. Notice that $\underline{U}^*, \underline{\Lambda}^*$ given by Proposition 12 and $\underline{V}^*$ given by Proposition 13 have the same form as U^*, Λ^* and V^* in Theorem 9. Hence, it remains to show that they are in fact the global minimizers of $\ell_{D,\sigma,h}$. Consider the following chain of inequalities.

$$\begin{aligned}
\ell_{D,\sigma,h}(\underline{U}^*, \underline{\Lambda}^*, \underline{V}^*) &\geq \ell_{D,\sigma,h}(U^*, \Lambda^*, V^*) \\
&= \min_{U,\Lambda} \min_{V} \ell_{D,\sigma,h}(U, \Lambda, V) \\
&\geq \min_{U,\Lambda} \underline{\ell}_{D,\sigma,h}(U, \Lambda) \quad \text{by Proposition 11} \\
&= \underline{\ell}_{D,\sigma,h}(\underline{U}^*, \underline{\Lambda}^*) \quad \text{by Proposition 12}
\end{aligned}$$

Due to Proposition 13 we know $\ell_{D,\sigma,h}(\underline{U}^*, \underline{\Lambda}^*, \underline{V}^*) = \underline{\ell}_{D,\sigma,h}(\underline{U}^*, \underline{\Lambda}^*)$. Hence the above chain of inequalities becomes a chain of equalities and specifically $\ell_{D,\sigma,h}(\underline{U}^*, \underline{\Lambda}^*, \underline{V}^*) = \ell_{D,\sigma,h}(U^*, \Lambda^*, V^*)$. Hence $\underline{U}^*, \underline{\Lambda}^*$ and $\underline{V}^*$ are also the global minimizers of $\ell_{D,\sigma,h}$. Rotating the result by R completes the proof and provides a minimizer of $\ell_{H,\sigma,h}$. □

3.2.2 Proof of Theorem 10

Recall that Theorem 10 addresses the minimization of the simplified objective function which we denote as follows

$$\ell(\lambda) := \begin{cases} \frac{(\sum_{i=1}^d D_i \lambda_i)^2}{4d\lambda_{max}} + \sigma^2 \sum_{i=1}^d \frac{1}{\lambda_i} + \sigma^2 \frac{1}{\lambda_{max}} d & \text{if } h^2 \mathbf{1} \succeq \lambda \succ 0 \\ \infty & \text{otherwise.} \end{cases}$$

Hence, our objective is to find the set of global minimizers given by $\mathcal{M}^* = \arg\min_\lambda \ell(\lambda)$.

As in the proof for Theorem 9, We assume that D is diagonal, increasing and $\sum_{i=1}^d D_i \geq 0$. For the current proof, we let K be the number of non-positive elements in D. Given the structure of D this implies that $D_i \leq 0$ for $1 \leq i \leq K$ and $D_i > 0$ for $K+1 \leq i \leq d$. If all elements are positive then $K = 0$.

We can immediately derive the first properties of the solution set $\mathcal{M}^*$.

Proposition 14. *Let* $\mathcal{M}' = \{\lambda \in \mathcal{M}^* : \lambda_1 = \max_{1\leq i\leq d} \lambda_i\}$ *be the set of all solutions with the first element being maximal. Then for each* $\lambda' \in \mathcal{M}'$ *we have the following properties*

1. λ' *is in monotonically decreasing order,*
2. *there exists* $K \leq J \leq d$ *such that* $\lambda'_i = h^2$ *for* $1 \leq i \leq J$ *and* $\lambda'_i < h^2$ *for* $J+1 \leq i \leq d$,
3. *if* $D_i = D_j$ *and* $2 \leq i, j \leq d$ *then* $\lambda'_i = \lambda'_j$.

Lastly, every other solution $\lambda^* \in \mathcal{M}^*$ *is a permutation of a* $\lambda' \in \mathcal{M}'$ *That is, for every* $\lambda^* \in \mathcal{M}^*$ *with* $j \in \arg\max_{1\leq i\leq d} D_i$ *and* $\lambda^*_j = \max_{1\leq i\leq d} \lambda^*_i$ *there exists* $\lambda' \in \mathcal{M}'$ *such that* $\lambda'_1 = \lambda^*_j$, $\lambda'_j = \lambda^*_1$ *and* $\lambda'_i = \lambda^*_i$ *when* $i \neq j$ *and* $2 \leq k \leq d$.

We first prove the following Lemma, showing that all eigenvalues corresponding to a non-positive diagonal element have to equal h^2.

Lemma 15. *Let* $\lambda^* \in \mathcal{M}^*$ *then* $\lambda^*_i = h^2$ *for* $1 \leq i \leq K$.

Proof. Assume for contradiction that for some $1 \leq i \leq K$ we have $\lambda_i < h^2$. Define the function $\lambda(\alpha) \in \mathbb{R}^d$ for $\alpha \in [0, 1]$ as

$$\lambda_i(\alpha) = \begin{cases} h^2 & \text{if } 1 \leq i \leq K \text{ or } \lambda_i + \alpha h^2 \geq h^2, \\ \lambda_i + \alpha h^2 & \text{otherwise.} \end{cases}$$

Given this definition $\lambda(\alpha) \succ \lambda$ for any chosen α value, including zero. Hence, the noise error component of the objective function will have strictly decreased. To complete the contradiction, we show that there exists an α^* for which the approximation error does not increase, that is

$$\frac{(\sum_{i=1}^d D_i \lambda^*_i)^2}{4d\lambda^*_{max}} \geq \frac{(\sum_{i=1}^d D_i \lambda^*_i(\alpha^*))^2}{4d\lambda^*_{max}(\alpha^*)}.$$

First, notice that $\lambda^*_{max}(\alpha^*) = h^2 \geq \lambda^*_{max}$, hence the denominator did not decrease. It remains to show that the numerator does not increase. By Lemma 39 and the assumption that $\sum_{i=1}^d D_i \geq 0$ we know that by optimality of λ^* that $\sum_{i=1}^d D_i \lambda^*_i \geq 0$. Due to $\lambda^*(0)$ increasing all elements

corresponding to negative diagonal elements, it must be that $\sum_{i=1}^{d} D_i\lambda_i^* \geq \sum_{i=1}^{d} D_i\lambda_i^*(0)$. It remains to show that this is true in magnitude also. If $\sum_{i=1}^{d} D_i\lambda_i^*(0) \geq 0$ then we are done. If $\sum_{i=1}^{d} D_i\lambda_i^*(0) < 0$ then by assumption $\sum_{i=1}^{d} D_i \geq 0$ and $\lambda^*(1) = h^2\mathbf{1}$ it follows that $\sum_{i=1}^{d} D_i\lambda_i^*(1) \geq 0 > \sum_{i=1}^{d} D_i\lambda_i^*(0) \geq 0$. Hence, by the intermediate value theorem there exists α^* such that $\sum_{i=1}^{d} D_i\lambda_i^*(\alpha^*) = 0$. Therefore, no λ^* for which the first K elements are not equal to h^2 can be a solution. □

Proposition 14. We first prove the last statement, namely that every $\lambda^* \in \mathcal{M}^*$ is a permutation of some $\lambda' \in \mathcal{M}^*$. Let $\lambda^* \in \mathcal{M}^*$ such that $k \in \arg\max_{1\leq i\leq d} \lambda_k^*$ and $\lambda_1^* < \lambda_k^*$. Due to Lemma 15 this can only happen when all elements of D are positive. Now, let $\lambda_k^* = \Delta + \lambda_1^*$ and consider the numerator of the approximation error

$$\begin{aligned} &D_1\lambda_1^* + D_k\lambda_k^* = D_1\lambda_1^* + D_k\Delta + D_k\lambda_1^* \\ &\geq D_1\lambda_k^* + D_k\lambda_1^* \end{aligned}$$

which is true since $D_1 \leq D_k$ by assumption. Since D has only positive elements, $D_1\lambda_k^* + D_k\lambda_1^* \geq 0$ and hence the numerator has not increased by swapping λ_k^* and λ_1^*. Let λ' be given by swapping the elements of 1 and k in λ^*. Then $\ell(\lambda') \leq \ell(\lambda^*)$ making $\lambda' \in \mathcal{M}'$.

Next, we prove Item 3 for $\lambda' \in \mathcal{M}'$. Assume for contradiction that when $D_i = D_j$ we have $\lambda_i \neq \lambda_j$ for $2 \leq i, j \leq d$. By strict convexity of $\frac{1}{x}$ we have that

$$\frac{1}{\lambda_i'} + \frac{1}{\lambda_j'} > \frac{2}{\frac{1}{\lambda_i'} + \frac{1}{\lambda_j'}}$$

which leads to reducing the noise error by replacing λ_i' and λ_j' by their average. Note that the approximation error does not change since the sum $D_i\lambda_i' + D_j\lambda_j'$ due to $D_i = D_j$ remains constant and the denominator remains unaffected since the maximal element, that is λ_1^*, is not modified.

We now consider Item 1. We know that λ_1' is maximal. Hence, for contradiction assume that

$\lambda'_i < \lambda'_j$ for $1 < i < j$. When $D_i = D_j$ then we already know by the above proof for Item 3 that this cannot happen. Then, let $D_i < D_j$ and by Lemma 15 let $D_i > 0$ as otherwise $\lambda'_i = h^2 \geq \lambda'_j$. Now, similar to before, let $\lambda'_j = \Delta + \lambda'_i$ with the following inequalities

$$\begin{aligned}\lambda'_i D_i + \lambda'_j D_j &= \lambda'_i D_i + \Delta D_j + \lambda'_i D_j \\ &> (\lambda'_i + \Delta) D_i + \lambda'_i D_j \\ &= \lambda'_j D_i + \lambda'_i D_j.\end{aligned}$$

Since $D_i, D_j > 0$ we have $\lambda'_j D_i + \lambda'_i D_j \geq 0$. This implies that we have strictly decreased the numerator of the approximation error by switching the positions of λ'_i and λ'_j. This leads to a decrease in ℓ, as swapping positions does not affect any other loss component.

We now complete the proof by considering Item 2. By Lemma 15 it is true that for some $K \leq J \leq d$ we have $\lambda'_i = h^2$. The fact that there do not exist $J < i < j \leq d$ such that $\lambda_i < \lambda_j = h^2$ follows from λ' being decreasing. □

Here, we see that the position of the elements is decreasing. Also, we see that every element associated with the maximal element of D has a solution associated with it. Without loss of generality, we can just assume that the maximal element occurs at the first index then. Based on this observation, we now introduce a simplified objective function, which we call the proxy function.

$$\ell^{(\text{proxy})}_{D,\sigma,h}(\lambda) := \begin{cases} \frac{(\sum_{i=1}^d D_i \lambda_i)^2}{4d\lambda_1} + \sigma^2 \sum_{i=0}^d \frac{1}{\lambda_i} + \sigma^2 \frac{1}{\lambda_1} d & 0 \succ \lambda \succeq h^2 \\ \infty & \text{otherwise.} \end{cases}$$

We denote the set of global solutions as $\mathcal{M}^{(\text{proxy})} = \arg\min_{\lambda \in \mathbb{R}^d} \ell^{(\text{proxy})}(\lambda)$. Letting $\Omega = \{\lambda : 0 \succ \lambda \succeq h^2\}$, it follows that constraining $\ell^{(\text{proxy})}$ to Ω we have differentiability $\ell^{(\text{proxy})}|_\Omega$.

With the following we want to state the seemingly obvious, that is all solutions of the proxy are also solutions of the true loss. Furthermore, all solutions in $\mathcal{M}'$ are solutions of the proxy. Hence, it will suffice finding the minima of $\ell^{(\text{proxy})}$.

Lemma 16. *It holds that* $\mathcal{M}^{(proxy)} \subseteq \mathcal{M}^*$ *with* $\mathcal{M}^{(proxy)} = \mathcal{M}'$.

Proof. Note that $l(\lambda) \leq \ell^{(\text{proxy})}(\lambda)$ for $\lambda \in \mathbb{R}^d$ and $l(\lambda) = \ell^{(\text{proxy})}(\lambda)$ whenever $\lambda_{max} = \lambda_1$. Consider $\lambda' \in \mathcal{M}'$. Since the first element is maximal it holds that $\ell^{(\text{proxy})}(\lambda') = \ell(\lambda') \leq \ell(\lambda) \leq \ell^{(\text{proxy})}(\lambda)$ for $\lambda \in \mathbb{R}^d$ and hence $\lambda' \in \mathcal{M}^{(\text{proxy})}$.

Next, we show that if $\lambda^{(proxy)} \in \mathcal{M}^{(\text{proxy})}$ then $\lambda^{(proxy)} \in \mathcal{M}'$. Since l is well defined, there exists some $\lambda' \in \mathcal{M}'$ and by the above $\lambda' \in \mathcal{M}^{(\text{proxy})}$. Since both $\lambda^{(proxy)}$ and λ' are then global minima for the proxy function it holds that $\ell^{(\text{proxy})}(\lambda') = \ell^{(\text{proxy})}(\lambda^{(proxy)})$. It remains to show that the first element $\lambda^{(proxy)}$ is also maximal. If the number of non-positive elements in D is greater than zero and hence $D_1 \leq 0$, then by an identical proof to Lemma 15 we have that $\lambda_1 = h^2$ and hence the first element is maximal. Otherwise, assume for contradiction that $k \in \arg\max_{1 \leq i \leq d} \lambda^{(proxy)}$ and $1 \notin \arg\max_{1 \leq i \leq d} \lambda^{(proxy)}$. Consider the numerator in the first term of the loss, since $D_1 \leq D_k$ and letting $\lambda_k^{(proxy)} = \lambda_1^{(proxy)} + \Delta$ we have

$$\begin{aligned} D_1\lambda_1^{(proxy)} + D_k\lambda_k^{(proxy)} &= D_1\lambda_1^{(proxy)} + D_k\lambda_1^{(proxy)} + D_k\Delta \\ &\geq D_1\lambda_k^{(proxy)} + D_k\lambda_1^{(proxy)} \geq 0. \end{aligned}$$

Hence, replacing $\lambda_1^{(proxy)}$ with $\lambda_k^{(proxy)}$ does not increase the numerator. Furthermore, since $\lambda_1^{(proxy)}$ occurs in the denominator of the first and third term, increasing it will strictly decrease the loss. Hence $\lambda_1^{(proxy)}$ has to be maximal. Since the first term is maximal it also holds that $l(\lambda') = \ell^{(\text{proxy})}(\lambda') = \ell^{(\text{proxy})}(\lambda^{(proxy)}) = \ell(\lambda^{(proxy)})$ and hence $\lambda^{(proxy)} \in \mathcal{M}'$. □

To find the global minimum of the proxy function, we may simply iterate through all constraint combination. For this, denote the working set by $\mathcal{W}$ which is a subset of $\{1, \ldots, d\}$. The complement $\mathcal{W}^c$ is such that $\mathcal{W} \cup \mathcal{W}^c = \{1, \ldots, d\}$ and $\mathcal{W} \cap \mathcal{W}^c = \emptyset$. We then introduce $\lambda \to \lambda^{\mathcal{W}}$ such that $\lambda_i^{(\mathcal{W})} = h^2$ for $i \in \mathcal{W}$ and $\lambda_i^{(\mathcal{W})} = \lambda_i$ for $i \in \mathcal{W}^c$. Now, for each working set we can find the set of candidate solutions, that is $\mathcal{T}^{(\mathcal{W})} = \{\lambda^{(\mathcal{W})} : \nabla \ell^{(\text{proxy})}|_\Omega(\lambda^{(\mathcal{W})}) = 0\}$. The minima of all candidate solutions for every working set will be the global minima of $\ell^{(\text{proxy})}$, as the set of all candidate solutions provides all the points satisfying the KKT conditions.

To better characterize the candidate solutions by relating them to the set of global minima for each working set given by $\mathcal{M}_{\mathcal{W}}^{(\text{proxy})}$. Specifically, the following proposition tells us that when a candidate solution exist, then it is the unique global minimum for the working set problem.

Lemma 17. *For every working set $\mathcal{W}$, we have $(\lambda')^{(\mathcal{W})}$ is the unique global minimum for $\ell^{(proxy)}(\lambda^{(\mathcal{W})})$ if $\nabla \ell^{(proxy)}|_{\Omega}((\lambda')^{(\mathcal{W})}) = 0$.*

Proof. To show the above, we simply show that the function $\ell^{(\text{proxy})}(\lambda^{(\mathcal{W})})$ is strictly convex. When $1 \in \mathcal{W}$ this is immediate since it is the sum of strictly convex functions. When $1 \notin \mathcal{W}$, it remains to show that $\frac{(\sum_{i=1}^{d} D_i \lambda_i^{(\mathcal{W})})^2}{4d\lambda_1^{(\mathcal{W})}}$ is convex. For this note the Hessian on the domain Ω can be written as

$$\nabla^2 \frac{(\sum_{i=1}^{d} D_i \lambda_i^{(\mathcal{W})})^2}{4d\lambda_1^{(\mathcal{W})}} = \left((\frac{1}{\sqrt{2d\lambda_1}})D - \sqrt{\frac{8d}{\lambda_1^3}}(\sum_{i=1}^{d} D_i \lambda_i) e_1 \right) \left((\frac{1}{\sqrt{2d\lambda_1}})D - \sqrt{\frac{8d}{\lambda_1^3}}(\sum_{i=1}^{d} D_i \lambda_i) e_1 \right)^T.$$

Hence, the Hessian is positive semi-definite and the term is convex. Since the other terms are strictly convex, $\ell^{(\text{proxy})}(\lambda^{(\mathcal{W})})$ is strictly convex. □

Now, we want to show that no global minimum exists that is not also a candidate solution. This will allow us to use the condition that the gradient vanishes to characterize any solution. Furthermore, we show with the following that it suffices to only check at most d rather than 2^d potential working sets. For this, define the working set $\mathcal{W}_J = \{1, \ldots, J\}$.

Proposition 18. *Let $K \leq J \leq d$ be the smallest integer such that $\nabla \ell^{(proxy)}|_{\Omega}((\lambda')^{(\mathcal{W}_J)}) = 0$ for some $(\lambda')^{(\mathcal{W}_J)}$. Then $(\lambda')^{(\mathcal{W}_J)}$ is the unique minimizer of $\ell^{(proxy)}$.*

Proof. By the KKT conditions, to find all the global minima of $\ell^{(\text{proxy})}$ it suffices to find all points at which the gradient vanishes given some working set and choose the ones inducing the minimal loss. By Proposition 14 it suffices to only consider working sets of the form $\mathcal{W}_J$ for $J \geq K$. Let $\mathcal{W}_J$ and $\mathcal{W}_{J'}$ be two working set for which the gradient vanishes at $(\lambda')^{(\mathcal{W}_J)}$ and $(\lambda')^{(\mathcal{W}'_J)}$. By Lemma 17 the points are the unique global minima for their respective functions. Let $J > J'$ and J be the smallest integer for which the above holds. Then $\ell^{(\text{proxy})}((\lambda')^{(\mathcal{W}_J)}) \leq \ell^{(\text{proxy})}((\lambda')^{(\mathcal{W}_{J'})})$ since $(\lambda')^{(\mathcal{W}_{J'})}$ is a feasible point under the working set $\mathcal{W}_J$. By uniqueness of global minima

for each working set, either $(\lambda')^{(\mathcal{W}_J)} = (\lambda')^{(\mathcal{W}'_J)}$ or $\ell^{(\text{proxy})}((\lambda')^{(\mathcal{W}_J)}) < \ell^{(\text{proxy})}((\lambda')^{(\mathcal{W}'_J)})$, making $(\lambda')^{(\mathcal{W}_J)}$ the unique global minimum. □

A priori it is not clear which working set will have a valid candidate solution. So, for each working set we derive the following necessary conditions for a point to be a valid candidate solution. Using these conditions we can construct a point and then if it is within the domain Ω it is a valid candidate solution.

Proposition 19. *Let $K \leq J \leq d$ then for $\ell^{(proxy)}(\lambda^{(\mathcal{W}_J)}) = 0$ it is necessary for $\lambda^{(\mathcal{W}_J)}$ to satisfy the following equations*

$$\lambda_1^{(\mathcal{W}_J)} = \begin{cases} h^2 & \text{if } J = 0, \\ \frac{2d}{aD_1}\left(\frac{a^2}{4d} + \sigma^2(d+1)\right) & \text{otherwise;} \end{cases}$$

$$\lambda_i^{(\mathcal{W}_J)} = \begin{cases} h^2 & \text{if } i \leq J, \\ \sigma\sqrt{\frac{2d\lambda_1^{(\mathcal{W}_J)}}{aD_j}} & \text{otherwise,} \end{cases} \qquad \text{for } 2 \leq i \leq d,$$

$$a = \sum_{i=J+1}^{d} D_i \lambda_i^{(\mathcal{W}_J)} + h^2 \sum_{i=1}^{J} D_i.$$

The result provides $d+1$ equations in as many unknowns. Due to the structure of the equations, if we can find a solution to a, then the other quantities are immediate. The following results provides the unique solution for a given some working set $\mathcal{W}$. Note that if a current working set $\mathcal{W}_J$ does not provide a candidate solution, then the a given below will make some of the equations infeasible. Specifically, the predicted $\lambda^{(\mathcal{W}_J)} \succ h^2\mathbf{1}$ which is not possible.

Proposition 20. *Let $K \leq J \leq d$, then for any candidate solution it follows that a is given by*

- *If $J = 0$ is empty, then we have*

$$a = \sqrt{2}\left(\frac{d\sigma^2}{D_1}\left(C\sqrt{8D_1(d+1) + C^2} + 2D_1(d+1) + C^2\right)\right)^{1/2}$$

with $C = \sum_{i=2}^{d} \sqrt{D_i}$;

- *Otherwise,*

$$a = \begin{cases} \frac{\sqrt[3]{\frac{2}{3}}c_2}{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}} + \frac{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}}{\sqrt[3]{18}} & \text{if } 27c_1^2 - 4c_2^3 \geq 0, \\ \cos(\theta/3)2\sqrt{\frac{c_2}{3}} & \text{otherwise,} \end{cases}$$

where $c_1 = \sigma\sqrt{\frac{dh^2}{2}}\sum_{i=J+1}^{d}\sqrt{D_i}$, $c_2 = \sum_{i=1}^{J} D_i h^2$ *and* $\theta = arccos(\frac{9c_1}{\sqrt{12c_2^3}})$.

The proofs for the above two propositions are technical and we defer them to Appendix 3.6.2.

Theorem 10. The first three statements follow directly from Proposition 14. The equations which provide the solution are given by Propositions 19 and 20. Lastly, every other solution in $\mathcal{M}^*$ being derived from λ^* follows from Proposition 14 and Lemma 17. By Proposition 14, $\mathcal{M}^*$ is made up of permutations of $\mathcal{M}'$. By Lemma 17, $\mathcal{M}'$ contains only one element. □

3.3 Error Rates and Geometry of Optimal Sample Sets

In this section we aim to provide a better understanding of what the optimal sample set achieves. We will focus both on the effect of the sample set on the loss ℓ as given in Definition 7. Furthermore, we focus on the geometry, that is the layout, of the sample set. This focus will highlight some of the key traits of optimal sample sets and provide an understanding of how such sets achieve the improved error rates.

To study the problem, we assume that some symmetric matrix H, a noise level $\sigma > 0$, and a constraint level $h > 0$ are given. As in the previous section, this will be based on a theoretical discussion in which each of these values is known exactly. This is meant to highlight what structure and properties optimal sample sets may display.

To simplify the discussion, as justified by Proposition 8 we assume that H is diagonal and given by the diagonal matrix of eigenvalues D. Notice that when discussing the standard basis vectors in this basis, these coincide with the eigenvalues of H. Furthermore, based on the same result,

we assume that D is in monotonically increasing order and that $\sum_{i=1}^{d} D_i \geq 0$. To study the properties of the sample set, we will divide the problem into cases where D is indefinite, that is, the diagonal includes strictly positive and negative elements. The other is the definite case, where all elements are positive.

3.3.1 Error Rates

We now consider the error rates of a theoretically optimal sample set compared to theoretically optimal forward differences. For this recall, the loss ℓ provides exactly the mean squared error for a quadratic function when $h = \infty$. When $h < \infty$, the constraint is aimed at avoiding higher-order errors which take effect with a larger sample set radius. As we will show in more detail later, the constraint also helps when we only have noisy estimates of the Hessian.

Before we consider the precise error rates, we consider the qualitative difference of the loss function induced by the structure of a sample set as given by Theorem 9 and one given by forward differences. For this recall the loss for a sample set is given by

$$\ell(S) = \begin{cases} \frac{1}{4}\|S^{-T} p_H(\mathcal{S})\|_2^2 + \sigma^2 \|S^{-1}\|_2^2 + \sigma^2 \|S^{-T}\mathbf{1}\|_2^2 & \text{if } hI \succeq S \succ 0 \\ \infty & \text{otherwise.} \end{cases}$$

Now, instead of using the optimality conditions from Theorem 9 we first consider the form when given a diagonal sample set, which is used for forward differences. For this we simply consider a diagonal matrix Λ giving

$$\ell(\Lambda) = \begin{cases} \frac{1}{4}\sum_{i=1}^{d} D_i^2 \lambda_i + 2\sigma^2 \sum_{i=1}^{d} \frac{1}{\lambda_i} & \text{if } hI \succeq \Lambda \succ 0 \\ \infty & \text{otherwise.} \end{cases} \tag{3.3.1}$$

Now, we observe how we transform $\ell(S)$ when using the optimality conditions from Theorem 9

for some diagonal Λ in the form of $\ell(U^*\Lambda(V^*)^T)$.

$$\ell(U^*\Lambda V^*) = \begin{cases} \frac{1}{4}\frac{(\sum_{i=1}^d D_i\lambda_i)^2}{d\lambda_{max}} + \sigma^2\sum_{i=0}^d \frac{1}{\lambda_i} + \sigma^2\frac{1}{\lambda_{max}}d & \text{if } hI \succeq \Lambda \succ 0 \\ \infty & \text{otherwise.} \end{cases} \tag{3.3.2}$$

The choice of V^* and U^* were crucial in this and both Remarks 2 and 3 provide a more intuitive explanation of how the choice of both led to this loss.

While we will consider a bound on this for an improved Λ, it is instructive to first focus on the qualitative differences between Equations 3.3.1 and 3.3.2. First, the loss is able to utilize the maximum singular value of S. Specifically, we divide both the approximation error and replace $\sum_{i=1}^d \frac{1}{\lambda_i}$ by $d\frac{1}{\lambda_{max}}$, providing an almost immediate improvement whenever not all elements in λ are equal. For the approximation error, the sum $\sum_{i=1}^d D_i^2\lambda_i$ is replaced by $\frac{(\sum_{i=1}^d D_i\lambda_i)^2}{d\lambda_{max}}$. Note, the division by d does not immediately provide an improved error, and is simply necessary to keep the right structure of the loss. That is, when all eigenvalues given by Λ are equal, then the two loss-functions have to coincide, which the division by d accomplishes. Notice that the form $(\sum_{i=1}^d D_i\lambda_i)^2$ allows for a key benefit not present in forward differences, namely when negative diagonal elements are present we can force the approximation error to be zero as we explore in Section 3.3.1.

Before we proceed, we note the optimal finite difference solution and bound its error.

Lemma 21. *The forward difference sample set given by* $\lambda^{(FD)} = \arg\min_{\Lambda\in\mathbb{R}^{d\times d},\Lambda diagonal}\ell(\Lambda)$ *has the form* $\lambda_i^{(FD)} = \min(h^2, \sqrt{8}\frac{\sigma}{|D_i|})$ *for* $1 \le i \le d$. *When* $\lambda^{(FD)} \preceq h^2$ *the bound on the loss is given by* $\ell(\lambda^{(FD)}) = \Theta(\sigma\sum_{i=1}^d |D_i|)$. *When* $\lambda^{(FD)} = h^2\mathbf{1}$ *then* $\ell(\lambda^{(FD)}) = \frac{h}{4}\sum_{i=1}^d |D_i| + 2\sigma^2\frac{d}{h}$.

Proof. The form is given by differentiating and setting to zero. The error rate is immediate by plugging in the result and simplifying. □

When $\lambda^{(FD)} \prec hI$ then $\ell(\lambda^{(FD)}) = \Theta(\sigma\sum_{i=1}^d |D_i|)$. Whenever either the noise level σ or any diagonal element D_i increases in magnitude, the error increases linearly. Furthermore, note the implicit dimensional dependence. If an element with the average of the absolute value

of D is added, then the error increases linearly with dimension. When $\lambda^{(FD)} = hI$, then $\ell(\lambda^{(FD)}) = \frac{h}{4}\sum_{i=1}^{d}|D_i| + 2\sigma^2\frac{d}{h}$. Here, when h is large the error will scale with $h^2\sum_{i=1}^{d}|D_i|$ while when h is small the error scales as $\sigma^2\frac{d}{h^2}$. The geometric properties will be contrasted with S^* in Section in Section 3.3.2.

In-Definite Case

Here we consider the case when D is in-definite, and has some positive and negative elements. As alluded to earlier, some specific properties of the loss can be exploited to remove the approximation error. Since using the exact optimal Λ^* to derive a bound is difficult, we use this insight to upper-bound the error that the optimal sample set achieves.

Proposition 22. *Let D be in-definite with $D_i \leq 0$ for $1 \leq i \leq K$ and $D_i > 0$ for $K+1 \leq i \leq d$. Let S^* be optimal for $\ell(S)$, then we have that*

$$\ell(S^*) = \mathcal{O}\left(\frac{d\sigma^2}{h^2}\frac{\sum_{m=K+1}^{d} D_m}{\sum_{j=1}^{K}|D_j|}\right).$$

Proof. We consider a heuristic choice of λ which is aimed to exploit the specific form of $\ell(U^*, \Lambda, V^*)$. For this, set $\lambda_i = h^2$ for $1 \leq i \leq K$ and $\lambda_i = h^2\frac{\sum_{j=1}^{K}|D_j|}{\sum_{m=K+1}^{d} D_m}$. We then have that $\sum_{i=1}^{d}\lambda_i D_i = 0$ and hence the approximation error is zero. We then have an upper-bound on the loss in terms of the noise error via $\ell(S^*) \leq \frac{\sigma^2}{h^2}\left(d + K + (d-K)\frac{\sum_{m=K+1}^{d} D_m}{\sum_{j=1}^{K}|D_j|}\right)$ completing the result. □

We briefly review the error bound. First, the loss scales as $\frac{1}{h^2}$ and the larger we can make the constraint the smaller the error is. Furthermore, note the term $\frac{\sum_{m=K+1}^{d} D_m}{\sum_{j=1}^{K}|D_j|} \geq 1$. The closer the sum of the negative diagonal elements is to the positive elements, the less this multiplier affects the loss. We have an explicit dimensional dependence in d. When adding positive diagonal elements, this effect is compounded and can be seen as a quadratic dimensional dependence. On the other hand, adding negative diagonal elements counteracts this. Lastly, we observe a quadratic noise dependence.

The most clear comparison to forward differences occurs when $\lambda^{(FD)} = h^2\mathbf{1}$. In the case of a small h, the error $\sigma^2\frac{d}{h}$ dominates. Hence, the errors appear identical except for $\frac{\sum_{m=K+1}^{d} D_m}{\sum_{j=1}^{K}|D_j|}$. When h is large we have that $h\sum_{i=1}^{d}|D_i|$ dominates, providing an error rate which does not depend on the noise level but increases linearly with the constraint. When $\lambda^{(FD)} \preceq h^2\mathbf{1}$ the differences are more pronounced. Depending on which terms are varied one error may be better than the other.

Importantly, the optimal sample set will have an error that is upper-bounded by minimum of both the above and forward differences. The key takeaway is that the loss function given by Theorem 9 allows for the immediate removal of the approximation error, leading to an error rate which can be made arbitrarily small when h grows and one that decreases when negative diagonal elements are increased in magnitude. This is in stark contrast to forward differences.

Definite Case

We now consider a definite cases, where $D \succ 0$. To simplify this case, we assume that $\lambda^* \prec h^2$. Since all diagonal elements are positive, this will always happen when h is large. Here we will focus on how dividing by the smallest singular value affects the error. Specifically, since λ_{max} is tightly related to the minimum diagonal element, we expect this quantity to be prominent in an error bound.

Proposition 23. *Let D be positive-definite and $S^* \preceq h^2$ be optimal for $\ell(S)$ as given in Definition 7. Then, we have that*

$$\ell(S^*) = \Theta\left(\sigma\sqrt{D_1}\frac{\sum_{i=2}^{d}\sqrt{D_i}}{\sqrt{d}}\right).$$

Proof. Recall from the previous section the definition of $C = \sum_{i=2}^{d}\sqrt{D_i}$. To simplify the proof we define additional constants. First, we define an average of C by $\bar{C} = \frac{C}{d-1}$. We also define a ratio of the smallest eigenvalue to the average via $\gamma = \frac{\sqrt{D_1}}{\bar{C}}$. The final quantity we define does not have a simple interpretation but greatly simplifies the algebra and is given by $M = 8D_1(d+1)$. This quantity can also be written as $M = 8(d+1)\gamma^2\bar{C}^2$. We can now rewrite the expressions for

a and λ_1 as given in Propositions 19 and 20 yielding $a = \sqrt{\frac{d\sigma^2}{D_1}((C+\sqrt{C^2+M})^2 - \frac{1}{2}M)}$ and $\lambda_1 = \frac{d\sigma^2}{2aD_1^2}(C+\sqrt{C^2+M})^2$.

Since the solution λ^* is finite and positive, we can plug it into the loss and use the defined quantities to simplify

$$\begin{aligned}\ell(S^*) &= \frac{a^2}{4d\lambda_1^*} + \sigma^2 \sum_{i=2}^{d} \frac{1}{\lambda_i^*} + \frac{\sigma^2(d+1)}{\lambda_1^*} \\ &= \sigma\sqrt{\frac{D_1}{d}}\sqrt{(C+\sqrt{C^2+M})^2 - \frac{1}{2}M}\left(\frac{3C+\sqrt{C^2+M}}{2(C+\sqrt{C^2+M})}\right).\end{aligned}$$

Notice that λ^* and a have been replaced with C and M. To proceed further we note the following relation which follows by multiplying the denominator on the right hand side and matching terms

$$\begin{aligned}((C+\sqrt{C^2+M})^2 - \frac{1}{2}M)\left(\frac{3C+\sqrt{C^2+M}}{2(C+\sqrt{C^2+M})}\right)^2 &= \frac{1}{M}(C^3\sqrt{C^2+M} - C^4) \\ &\quad + C\sqrt{C^2+M} + \frac{5}{2}C^2 + \frac{M}{8}.\end{aligned}$$

To simplify the above we note two further relations $\frac{M}{C^2} = \frac{8(d+1)\gamma^2}{(d-1)^2}$ and $\frac{D_1}{M} = \frac{1}{8(d+1)}$. Combining these we aim to remove all instances of M and obtain the following simplification

$$\begin{aligned}\ell(S^*) &= \sigma\sqrt{\frac{D_1}{d}}\sqrt{\frac{1}{M}(C^3\sqrt{C^2+M} - C^4) + C\sqrt{C^2+M} + \frac{5}{2}C^2 + \frac{M}{8}} \\ &= \sigma\sqrt{\frac{D_1}{dM}}\sqrt{C^4(\sqrt{1+\frac{M}{C^2}} - 1) + MC^2\left(\sqrt{1+\frac{M}{C^2}} + \frac{5}{2} + \frac{M}{8C^2}\right)} \\ &= \sigma\sqrt{\frac{\bar{C}^4(d^-)^4}{8d(d^+)}}\sqrt{(\sqrt{1+\frac{8(d^+)\gamma^2}{(d^-)^2}} - 1) + 8\frac{(d^+)}{(d^-)^2}\gamma^2\left(\sqrt{1+\frac{8(d^+)\gamma^2}{(d^-)^2}} + \frac{5}{2} + \frac{(d^+)\gamma^2}{(d^-)^2}\right)}\end{aligned}$$

where we used $d^+ = d+1$ and $d^- = d-1$ to keep notation tight.

To provide a lower and upper-bound on this, recall that $0 \leq \gamma \leq 1$ and note that $\frac{d+1}{(d-1)^2} \leq 3$ for

$d \geq 2$. Now, notice the following bounds on the individual terms of the expression

$$\frac{7}{2} \leq \left(\sqrt{1 + \frac{8(d+1)\gamma^2}{(d-1)^2}} + \frac{5}{2} + \frac{(d+1)\gamma^2}{(d-1)^2} \right) \leq 21/2$$
$$0 \leq (\sqrt{1 + \frac{8(d+1)\gamma^2}{(d-1)^2}} - 1) \leq \frac{1}{2} \frac{8(d+1)\gamma^2}{(d-1)^2}$$

where the upper-bound for the last bound comes from $\sqrt{1+x}$ being concave. These bounds lead to

$$2\sqrt{7\frac{(d+1)}{(d-1)^2}\gamma^2} \leq \sqrt{(\sqrt{1 + \frac{8(d+1)\gamma^2}{(d-1)^2}} - 1) + 8\frac{(d+1)}{(d-1)^2}\gamma^2 \left(\sqrt{1 + \frac{8(d+1)\gamma^2}{(d-1)^2}} + \frac{5}{2} + \frac{(d+1)\gamma^2}{(d-1)^2} \right)}$$
$$\leq 2\sqrt{23\frac{(d+1)\gamma^2}{(d-1)^2}}.$$

Now, note the bounds on the various dimension dependent fractions given by $\frac{1}{d} \leq \frac{d+1}{(d-1)^2} \leq \frac{6}{d}$ and similarly $d^2\frac{1}{24} \leq \frac{(d-1)^4}{d(d+1)} \leq d^2$. Hence, we have that the loss is given by $\ell(S^*) = \Theta(\sigma \bar{C}^2 \sqrt{d}\gamma) = \Theta\left(\sigma\sqrt{D_1}\frac{\sum_{i=2}^d \sqrt{D_i}}{\sqrt{d}}\right)$. □

We can directly compare the error rate of $\mathcal{S}^*$ given by $\Theta\left(\sigma\sqrt{D_1}\frac{\sum_{i=2}^d \sqrt{D_i}}{\sqrt{d}}\right)$ to the one given by forward differences in $\Theta(\sigma \sum_{i=1}^d |D_i|)$. The similarity is that the noise dependence remains the same. However, crucially the error bound now depends on the product of $\sqrt{D_1}$ and $\sum_{i=2}^d \sqrt{D_i}$. This immediately indicates that a smaller D_1, irrespective of the other diagonal elements, leads to a smaller error. As seen in Theorem 10, a small D_1 leads to a large λ_1 element and hence a larger divisor in the loss function given by Equation 3.3.2. Another benefit is that if any diagonal element is increased individually, the error scales as the square root, rather than linearly as in forward differences. Lastly, we note the improved dimensional dependence due to the division by $\sqrt{d}$. Interestingly, even if all the element are the same, the structure of the optimal sample set retains this benefit in dimensional dependence. Intuitively, this is again because of being able to manipulate the interaction of λ_1 with the remaining singular values to reduce the overall error. We expand slightly further on this in Remark 4. Notably, if the constraint in h is tight, then some of this benefit will be lost.

3.3.2 Geometry of Sample Set

We now focus on the geometric properties of the sample set. This will help us understand what structure in the sample set leads to the improvements observed in the previous section. We first observe the following baseline result. This tells us what structure is optimal when we are fully agnostic to the approximation error. This happens in the presence of no prior information about the function.

Lemma 24. *The noise error given by* $\|S^{-1}\|_F^2 + \|S^{-T}\mathbf{1}\|^2$ *with a constraint* $0 \prec S \preceq hI$ *is minimized if and only if* $S^* = hU$ *where* U *is any orthogonal matrix.*

Proof. We have that for the second term, that is $\|S^{-1}\mathbf{1}\|^2$, to be minimized we simply need $\mathbf{1}$ to be a right singular vector associated with the largest eigenvalue. Now, we have that the first term is equal to $\sum_{i=1}^{d} \frac{1}{\lambda_i}$ by matrix norm properties. This is clearly minimized when each λ_i is maximal, that is $\lambda_i = h^2$ for $1 \leq i \leq d$. This constrains the sample set to be $S = hU$ for U being orthogonal. Notice, that this also minimized the second term as $\|S^{-1}\mathbf{1}\|^2 \geq \frac{1}{\lambda_{max}} d$. This is minimized when $\lambda_{max} = h^2$ also. Since for hU each singular value is given by h we have achieved the minimum. □

The optimal sample set without any information on the function is orthogonal. Note that this is similar to forward differences, and hence it often suffices to discuss an orthogonal geometry. We can now contrast such a structure with the optimal sample set which uses exact information of the Hessian. Contrasting both extremes provides insight into how a sample set may be structured in practice, when information is present but imprecise.

To characterize the geometry of the sample set, for each difference vector we consider the length, that is $\|s_i\|$, and the angle with each standard basis vector, that is $\frac{1}{\|s_i\|}|\langle e_k, s_i\rangle|$. Recall, since we are working in the eigenbasis of H, the angle formed with the standard basis is equivalent to the angle with each eigenvector.

Proposition 25. *Let* S^* *be the optimal sample set as given by Theorem 9. Then we have the following geometric results*

- *the norm is* $\|s_i^*\| = \sqrt{\frac{1}{d}\sum_{j=1}^{d}\lambda_j^*}$ *for* $1 \leq i \leq d$,
- *the angle to each coordinate direction is given by* $\frac{1}{\|s_i^*\|}|\langle e_k, s_i^*\rangle| = \frac{1}{\sqrt{\sum_{j=1}^{d}\lambda_j}}\sqrt{\lambda_k^*}$ *for* $1 \leq k, i \leq d$.

Proof. This follows immediately from plugging in $s_i^* = U^*\Lambda^*(V^*)^T e_i$ and noticing that $|s_i^*| = \frac{1}{\sqrt{d}}\lambda$. □

We observe that both the norm and angle do not depend on the difference vectors. That is, every difference vector will have the same length and span the same angle to each standard basis vector. This first result is in contrast to forward differences, where each difference vector has a different length. This result is chiefly due to the use of V^* being a Hadamard matrix, as the contrast in difference vectors is purely in the sign of each entry rather than the size. Hence, it enforces symmetry in the geometry of the sample set.

Another consequence of using the Hadamard matrix, and its induced symmetry, is in interpreting how the error of the gradient estimate is presented. For this, note that $p(\mathcal{S}^*) = (\sum_{i=1}^{d} D_i\lambda_i)\mathbf{1}$ which says that each $(s_i^*)^T D s_i^*$ is on the same level-set of the quadratic form given by D. We will see a visualization of this in Figure 3.1. This observation leads to the following result regarding errors of directional derivatives.

Lemma 26. *Let* $f(x)$ *be a second-order polynomial and let* S^* *be the optimal sample set for the simplex gradient estimator at* x_0. *Then the mean squared error of the forward-difference approximation* $\frac{\tilde{f}(x_0+s_i^*)-\tilde{f}(x_0)}{\|s_i\|}$ *of the directional derivative* $\partial_{s_i} f(x_0)$ *has the same mean squared error for* $1 \leq i \leq d$.

Proof. The error of a direction derivative estimate is given by

$$\begin{aligned}\mathbb{E}[|\frac{\tilde{f}(x_0+s_i^*)-\tilde{f}(x_0)}{\|s_i^*\|} - \partial_{s_i^*} f(x_0)|^2] &= |\frac{f(x_0+s_i^*)-f(x_0)}{\|s_i^*\|} - \partial_{s_i^*} f(x_0)|^2 + 2\frac{\sigma^2}{\|s_i^*\|^2} \\ &= \frac{1}{4}\left(\frac{(s_i^*)^T H s_i^*}{\|s_i^*\|}\right)^2 + 2\frac{\sigma^2}{\|s_i^*\|^2}\end{aligned}$$

where the last step follows from $f(x)$ being a quadratic polynomial. Clearly, since S^* is chosen such that $p(\mathcal{S}^*) = (\sum_{i=1}^{d} D_i\lambda_i)\mathbf{1}$ we have that $(s_i^*)^T H s_i^* = (s_j^*)^T H s_j^*$ for $1 \leq i, j \leq d$. Furthermore, by Proposition 25 we have that the norm of each difference vector is the same. Hence, the mean squared errors are the same for $1 \leq i \leq d$. □

Note that the simplex gradient can be seen as a change of basis on a set of d directional derivatives. Then, when disregarding higher-order errors, every directional derivative given by s_i has the same mean squared error. This is because the mean squared error in Definition 7 is equivalent to the error of a gradient estimate on a quadratic approximating a function locally. Hence, we can think of them each directional derivative contributing the same error to the simplex gradient approximation.

In-Definite Case

Just as in Section 3.3.1, it is difficult to provide a precise analysis of the indefinite case. Instead, we use a similar heuristic scheme to approximate the optimal sample set. This scheme ensures that the approximation error is removed, leaving only the noise error.

Proposition 27. *Let* $\lambda_i = h^2$ *for* $1 \leq i \leq K$ *and* $\lambda_i = h^2 \frac{\sum_{j=1}^{K} |D_j|}{\sum_{m=K+1}^{d} D_m}$ *for* $K+1 \leq i \leq d$. *We have that*

- *the norm grows as* $\|s_i\| = \Theta\left(h\sqrt{\frac{\sum_{j=1}^{K} |D_j|}{\sum_{m=K+1}^{d} D_m}}\right)$,
- *for* $1 \leq k \leq K$ *that* $\left|\frac{1}{\sqrt{K}} - \frac{1}{\|s_i\|}|\langle e_k, s_i\rangle|\right| = \mathcal{O}\left(\frac{d-K}{K^{3/2}}\sqrt{\frac{\sum_{j=1}^{K} |D_j|}{\sum_{m=K+1}^{d} D_m}}\right)$,
- *and for* $K+1 \leq k \leq d$ *that* $\left|\frac{1}{\sqrt{d}} - \frac{1}{\|s_i\|}|\langle e_k, s_i\rangle|\right| = \mathcal{O}\left(\frac{K}{d^{3/2}}\left(\frac{\sum_{m=K+1}^{d} D_m}{\sum_{j=1}^{K} |D_j|} - 1\right)\right)$

where the results hold for $1 \leq i \leq d$.

Proof. The first result follows immediately from $\|s_i\| = h\sqrt{\frac{K}{d} + \frac{(d-K)}{d}\frac{\sum_{j=1}^{K} |D_j|}{\sum_{m=K+1}^{d} D_m}}$.

For the next result consider $1 \leq k \leq K$ and $1 \leq i \leq d$ giving

$$\begin{aligned}\Big|\frac{1}{\sqrt{K}} - \frac{1}{\|s_i\|}|\langle e_k, s_i\rangle|\Big| &= \Big|\frac{1}{\sqrt{K}} - \frac{1}{\sqrt{K + (d-K)\frac{\sum_{j=1}^{K}|D_j|}{\sum_{m=K+1}^{d} D_m}}}\Big| \\ &\leq \frac{1}{2}\frac{d-K}{K^{3/2}}\frac{\sum_{j=1}^{K}|D_j|}{\sum_{m=K+1}^{d} D_m}\end{aligned}$$

where the last step follows from $\frac{1}{\sqrt{1+x}}$ is a convex function.

We proceed similarly for the last result where $K+1 \leq k \leq d$ and $1 \leq i \leq d$ giving

$$\begin{aligned}\Big|\frac{1}{\sqrt{d}} - \frac{1}{\|s_i\|}|\langle e_k, s_i\rangle|\Big| &= \Big|\frac{1}{\sqrt{d}} - \frac{1}{\sqrt{K(\frac{\sum_{m=K+1}^{d} D_m}{\sum_{j=1}^{K}|D_j|} - 1) + d}}\Big| \\ &\leq \frac{1}{2}\frac{K}{d^{3/2}}(\frac{\sum_{m=K+1}^{d} D_m}{\sum_{j=1}^{K}|D_j|} - 1).\end{aligned}$$

□

The norm will increase with h. This is intuitive, as once the approximation error is removed, the sample set is made as large as possible to reduce the noise error. The next results states that the differences vectors cluster around the directions corresponding to negative diagonal elements when the relative size of negative elements is small. This occurs since in the heuristic scheme, more weight has to be place on negative weights to remove the approximation error. On the other hand, in the next result we see that as the relative size of the weights is close to one, the difference vectors are now equally spread out.

An orthogonal sample set is then similar to the heuristic scheme when the relative size of the diagonal elements is close to one. However, crucially the rotation of the heuristic sample set is such that the approximation error is removed, while this is not the case for a generic orthogonal matrix. Now, when the size of positive elements is much larger than the negative elements, the matrix will be ill-conditioned and cluster around the K non-positive diagonal elements. While the optimal sample set will be less extreme than this heuristic, this is an obvious difference where the geometry is altered in response to the information gained about the Hessian. In

Section 3.5.1 we will briefly discuss how such overfitting can be addressed when the Hessian is inexact.

Definite Case

We now consider the case when $D \succ 0$. As in Section 3.3.1, we are able to provide a precise analysis when assuming that $\lambda^* \prec h^2$, that is the sample set is not affected by the constraints. While this will not always valid, in situations where the function is well approximated by a quadratic locally or the constraint can be made large enough, this is a valid assumption. This also provide intuition by considering an idealized case.

Proposition 28. *Let D be positive-definite and $\lambda^* \prec h^2$. Then following geometric properties on the difference vectors hold*

- *the difference vector size is given by* $\|s_i\| = \Theta\left(\sqrt{\frac{\lambda_1^*}{d}}\right) = \Theta\left(\sqrt{\sigma \sum_{m=2}^{d} \sqrt{D_m}(\frac{1}{dD_1^3})^{1/4}}\right)$,

- *the angle to the first standard basis vector is given by* $\left|1 - \frac{1}{\|s_i\|}|\langle e_1, s_i\rangle|\right| = \mathcal{O}\left(\frac{D_1}{d}\sum_{m=2}^{d}\frac{1}{D_m}\right)$

for $1 \leq i \leq d$.

To prove the above proposition we present three Lemmas which bound the size of various quantities. First we need to understand the growth of a central quantity to the solution λ^*.

Lemma 29. *Let a be as given in Theorem 10. Then its growth is given by* $a = \Theta\left(\sigma \bar{C} d^{3/2}\sqrt{\frac{1}{D_1}}\right)$.

Proof. Recall that $a = \sqrt{2\left(\frac{d\sigma^2}{D_1}(C\sqrt{8D_1(d+1)+C^2} + 2D_1(d+1) + C^2)\right)}$. We have an immediate lower-bound by removing occurrences of D_1 in the following

$$C\sqrt{8D_1(d+1)+C^2} + 2D_1(d+1) + C^2 \geq 2C^2.$$

For the upper-bound, first notice that $C^2 \geq (d-1)^2 D_1$, since $D_1 \leq D_i$ for $2 \leq i \leq d$ by assumption. Also, note that $\frac{d+1}{(d-1)^2} \leq 3$ for $d \geq 2$. Hence, we have

$$C\sqrt{8D_1(d+1)+C^2} + 2D_1(d+1) + C^2 \leq C\sqrt{8C^2\frac{d+1}{(d-1)^2} + C^2} + 2C^2\frac{d+1}{(d-1)^2} + C^2$$
$$\leq C\sqrt{24C^2 + C^2} + 6C^2 + C^2 = 12C^2.$$

□

Next, we consider the growth of the largest singular value.

Lemma 30. *Let λ^* be the solution as given by Theorem 10. We then have that $\lambda_1^* = \Theta\left(\sigma\bar{C}\left(\frac{d}{D_1}\right)^{3/2}\right)$.*

Proof. Recall that λ_1^* is given by $\lambda_1^* = \frac{2d}{aD_1}\left(\frac{a^2}{4d} + \sigma^2(d+1)\right)$. Based on Lemma 29 there exist $c^{(ub)}$ and $c^{(lb)}$ such that the following bounds hold

$$c^{(lb)}\frac{\sigma}{(d-1)\bar{C}}\sqrt{\frac{d}{D_1}}\left(c^{(ub)}\frac{(d-1)^2\bar{C}^2}{D_1}\right) \leq \lambda_1^* \leq c^{(ub)}\frac{\sigma}{(d-1)\bar{C}}\sqrt{\frac{d}{D_1}}\left(c^{(lb)}\frac{(d-1)^2\bar{C}^2}{D_1} + (d+1)\right).$$

Since $\bar{C}^2 \geq D_1$ we have $\frac{(d-1)^2\bar{C}^2}{D_1}$ dominates $d+1$ completing the result. □

Lastly, we require the relative size of the largest singular value to all the other ones. The result relies on combining the above two Lemmas.

Lemma 31. *Let λ^* be the solution as given by Theorem 10. We then have the following rate on the sum of fractions given by $\sum_{i=2}^d \frac{\lambda_i^*}{\lambda_1^*} = \Theta\left(\frac{D_1}{d\bar{C}}\sum_{i=2}^d \frac{1}{\sqrt{D_i}}\right) = \mathcal{O}(1)$. Additionally, we have the rate on each individual fraction given by $\frac{\lambda_i^*}{\lambda_1^*} = \Theta\left(\frac{D_1}{d\bar{C}}\sqrt{\frac{1}{D_i}}\right) = \mathcal{O}\left(\frac{1}{d}\right)$.*

Proof. We have that $\sum_{i=2}^d \frac{\lambda_i^*}{\lambda_1^*} = \frac{1}{2}\sigma\sqrt{\frac{2d}{a\lambda_1^*}}\sum_{i=2}^d \frac{1}{\sqrt{D_i}}$ and using Lemma 29 and 30 to bound a and λ_1^* we obtain the result. The result for $\frac{\lambda_i^*}{\lambda_1^*}$ follows similarly. □

We are now ready to provide the proof of the main results.

Proposition 28. We first consider the norm of each difference vector given by $\|s_i\| = \sqrt{\frac{1}{d}\sum_{i=1}^{d}\lambda_i^*}$. Notice that we can write $\|s_i\| = \sqrt{\frac{\lambda_1^*}{d}}\sqrt{1+\sum_{i=2}^{d}\frac{\lambda_i^*}{\lambda_1^*}}$. The lower-bound is immediate and the upper-bound follows from Lemma 3.3.2, leading to $\|s_1\| = \Theta(\sqrt{\frac{\lambda_1^*}{d}})$.

We now prove the result on the angle to r_1. For this we use the fact that the function $\frac{1}{\sqrt{1+x}}$ is convex to obtain

$$\left|1 - \frac{1}{\|s_i\|}|\langle e_k, s_i\rangle|\right| = \left|1 - \frac{1}{1+\sqrt{\sum_{i=2}^{d}\frac{\lambda_i^*}{\lambda_1^*}}}\right| \leq \frac{1}{2}\sum_{i=2}^{d}\frac{\lambda_i^*}{\lambda_1^*}.$$

Using Lemma 3.3.2 we have that

$$\left|1 - \frac{1}{\|s_i\|}|\langle e_k, s_i\rangle|\right| = \mathcal{O}\left(\frac{D_1}{d\bar{C}}\sum_{i=2}^{d}\frac{1}{\sqrt{D_i}}\right).$$

To obtain a simplified version we make two observations. First, since $\bar{C}$ is an average, by Jensen's Inequality we have that $\frac{1}{\bar{C}} = \frac{1}{\frac{1}{d-1}\sum_{i=2}^{d}\sqrt{D_i}} \leq \frac{1}{d-1}\sum_{i=2}^{d}\frac{1}{\sqrt{D_i}}$. Additionally, due to standard variance and mean properties, that is $\mathbb{V}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2 \geq 0$, we have that $\left(\frac{1}{d-1}\sum_{i=2}^{d}\frac{1}{\sqrt{D_i}}\right)^2 \leq \frac{1}{d-1}\sum_{i=2}^{d}\frac{1}{D_i}$ completing the proof. □

Remark 4. *We have that* $\frac{\lambda_i^*}{\lambda_1^*} = \Theta(\frac{D_1}{d\bar{C}}\sqrt{\frac{1}{D_i}})$. *When all diagonal elements are the same we have that* $\frac{\lambda_i^*}{\lambda_1^*} = \Theta(\frac{1}{d})$, *hence the largest* λ^* *will not equal all the others. Subsequently, since the angle is given by* $\frac{1}{\|s_i\|}|\langle e_k, s_i\rangle| = \frac{1}{\sqrt{\sum_{i=1}^{d}\lambda_i^*}}\sqrt{\lambda_1^*}$ *we also know that the sample set will not have a fully orthogonal structure once we pass some dimension—in practice we find this to be already the first dimension. This is intuitive as* λ_1^* *ought to be as large as possible, since we are dividing by it. So we sacrifice some noise error component to achieve a lower error by making* λ_1 *larger and hence dividing by a larger quantity in the loss.*

We first consider the size of the difference vectors. In its simplified form, it scales as the square root of the ratio of the largest eigenvalue and the dimension. It is notable here that the largest eigenvalue dominates as we also discuss in the above remark. When expanding, we observe that as expected, the size also scales with the noise-level. A larger noise-level leads to a large size of the sample set to mitigate the noise error. However, unlike forward differences, the size also

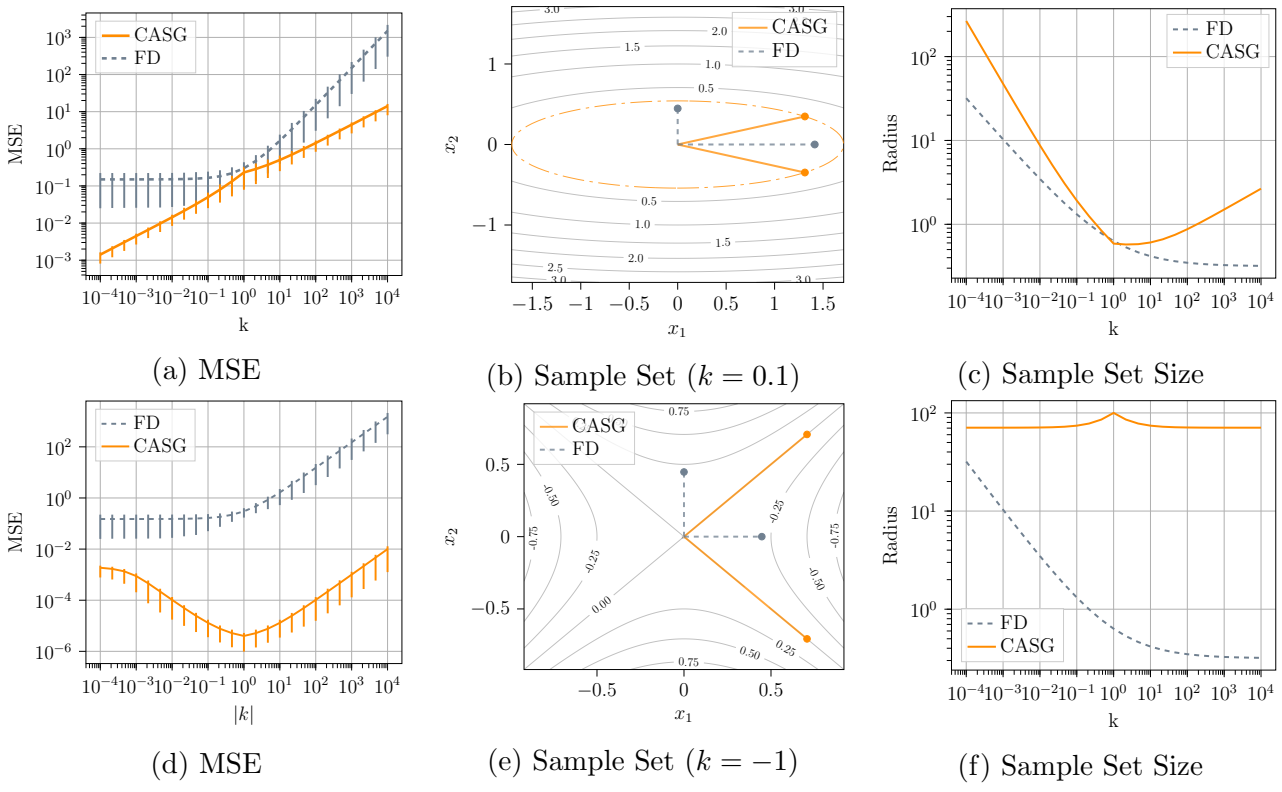
scales positively when increasing all diagonal elements except the first one. This is because these terms appear in the numerator of the approximation error and the first eigenvalue has to increase in response to reduce the effect of this. Lastly, as expected when the first diagonal element increases, the eigenvalue decreases as a smaller size is require to mitigate the increased approximation error.

Next, we focus on the angle to the first standard basis vector. The bound is given by the product of D_1 with the average of inverses, that is $\frac{1}{d}\sum_{i=2}^{d}\frac{1}{D_i}$. The difference vectors are then tightly clustered around e_1 when D_1 is small or the other diagonal elements are large, making the average of inverse small also. From the perspective of gradient estimation, this means that the difference vectors are aligned with the lowest curvature direction. The purpose of this is, that in the direction of lowest curvature, the difference vector lengths can be increased without enquiring a large approximation error. This is because in the direction of the smallest curvature, the gradient will change most slowly. By being able to increase the difference vectors, we can then mitigate the noise error further. We see this also highlighted in the sizes of difference vectors. We observed that these depend primarily on the size of the first eigenvalue, which in turn is highly depended on the relative size of the lowest curvature.

3.3.3 Example: Quadratic Function

We now consider the idealized scenario of estimating a gradient for a quadratic function for which the Hessian is given. This is to complement the theoretical results on the properties of the optimal sample set above. In this Section we refer to estimating the gradient based on the results of Section 3.2 as the Curvature Aligned Simplex Gradient (CASG)—in the following Section we will provide a breakdown of the algorithm.

The set-up is as follows. We estimate the gradient at the origin of a quadratic function corrupted by additive noise. Specifically, we consider the function $\tilde{f}_k(x, y) = kx^2 + y^2 + \epsilon$, where $k \in \mathbb{R}$ is used to control the conditioning of the Hessian and $\epsilon \sim \mathcal{N}(0, 0.01^2)$. We provide the exact Hessian and noise level to the gradient estimation methods. The mean squared error is given by the objective function in Definition 7 by construction. To obtain error bars for the mean



(a) MSE (b) Sample Set ($k = 0.1$) (c) Sample Set Size

(d) MSE (e) Sample Set ($k = -1$) (f) Sample Set Size

Figure 3.1: Comparing forward differences (FD) with optimal difference vector lengths [BFB15] to the curvature aligned simplex gradient (CASG). The first row corresponds to a positive definite Hessian with $k \in [10^{-4}, 10^4]$. The bottom row represents an indefinite Hessian with $k \in [-10^4, -10^{-4}]$ and with $h = 10^2$.

squared error plots in Figures 3.1a and 3.1d we perform 1,000 gradient estimates and plot the 25th and 75th percentile values. While for a quadratic function the mean squared error loss given in Definition 7 is exact, we set $h = 100$. While for the definite case we will have $S^* \prec hI$, for the indefinite case this is necessary as otherwise the error would vanish by Proposition 22 and the difference vectors would be made infinitely large as given in Proposition 27.

The observations on error rates are given in Figures 3.1a and 3.1d. We see that in both scenarios `CASG` is able to achieve significantly improved error rates compared to forward differences. For the definite case, as predicted in Proposition 23 when the condition number is high then the error rate can be vastly improved compared to FD. This is due to the error scaling directly with the lowest Hessian eigenvalue. For the indefinite case, we have an overall improvement, since the approximation error can be removed or made very small due to the improved form of the loss function as given by Equation 3.3.2. As also given by Proposition 22, when the condition number is close to unity the error is maximally reduced.

We now observe the geometric properties of the sample sets in the remaining Figures. We first note the geometric structure via visualization in Figures 3.1b and 3.1e. We observe that the difference vectors are located on the level sets of the quadratic $x^T Dx$ as given by Proposition 26. This indicates that the directional derivative along each difference vectors has the same error. Then note that in Figure 3.1b the differences vectors are along the lowest curvature direction as predicted by Proposition 28. This is in contrast to forward difference which is constrained along the standard basis directions and can not exploit low curvature directions. Note in the indefinite case, the differences vectors are placed along the zero level-set of the function and hence incur no approximation error loss. In Proposition 27 this was predicted by stating that all differences vectors have the same angle to the standard basis vectors when the negative and positive diagonal elements have the same size. Even though both forward differences and `CASG` are orthogonal in this specific case, the rotation of FFD does not allow it to exploit this structure.

Lastly, in Figures 3.1c and 3.1f we note the size of both methods by taking the average norm of the difference vectors. Note that as given by Proposition 25 for CASG, each difference vector has the same norm and hence is equal to the average. Clearly for both cases, `CASG` is able to achieve a large size and intuitively mitigate the noise error better. In the definite case, `CASG` achieves this by exploiting the low curvature directions directly by aligning the difference vectors appropriately. In the indefinite case this happens by aligning the difference vectors around the zero level-sets of the function, done by choosing the singular values in such a way as to cancel out the approximation error in Equation 3.3.2. These results align with the Geometric results in Propositions 28 and 27.

3.4 Algorithm: Curvature Aligned Simplex Gradient

We now introduce the *Curvature Aligned Simplex Gradient* (CASG) based on our theoretical results in Section 3.2. The naming follows from the intuition presented in Section 3.3 that differences vectors align with low curvature direction. We first present the base Algorithm 1

which assumes that a Hadamard matrix exists for the given dimension. Then we discuss the algorithm to obtain the optimal singular values based on Theorem 10. As the base algorithm does not yet address instances where no Hadamard matrix exists for the given dimension, we then provide a heuristic methods based on partition the space. We conclude with a runtime analysis of the methodology.

3.4.1 CASG: Base Algorithm

Algorithm 1 is based on Theorem 9 and describes the procedure to construct the minimzer S^* of the objective function given in Definition 7. The third line diagonalizes the input H. Simultaneously, without loss of generality it enforces that D has positive trace by negating if necessary as justified by Proposition 8. By Theorem 9, we have that $U^* = I$ leaving only the computation of Λ^* and V^*. We obtain Λ^* via the function call Get$\Lambda^*(D, \sigma, h)$ described further in the next Section. In line five we then obtain a Hadamard matrix. Notably, Sylvester's Construction allows us to obtain Hadamard matrices for dimensions which are power's of two. A current conjecture states that for all multiples of four a Hadamard matrix exists. The first dimension which is a multiple of four for which such matrix has not been found yet is 668 [KTR05] — a dimension too large to worry about in many practical applications. We discuss the practicalities in Section 3.4.3. Lastly, we pre-multiply the result by R to transform the solution back into the appropriate basis.

Algorithm 1 CASG

1: **procedure** COMPUTE $S^*(H, \sigma, h)$
2: $d \leftarrow$ Dimension of H
3: $R, D \leftarrow$ Diagonalize H, with D increasing order and positive trace
4: $\Lambda^* \leftarrow$ Get$\Lambda^*(D, \sigma, h)$ ▷ In descending order.
5: $V^* \leftarrow$ Hadamard matrix with first row all ones
6: $S^* \leftarrow$
7: **return** S^*
8: **end procedure**

3.4.2 Getting Λ^*

Algorithm 2 GetΛ^*

procedure GET$\Lambda^*(D, \sigma, h)$
  $d \leftarrow$ Dimension of D
  if $\sum_{i=1}^{d} D_i = 0$ **then**
    return h
  end if
  $J \leftarrow |\{i | D_i \leq 0, 1 \leq i \leq d\}|$
  $c_1, c_2 \leftarrow \sum_{i=J+1}^{d} \sqrt{D_i}, \sum_{i=1}^{J} D_i$
  while True **do**
    $\lambda_{J+1}, a \leftarrow \text{Get}\lambda_{J+1}(J, D, \sigma, h, c_1, c_2)$
    $c_1, c_2 \leftarrow c_1 - \sqrt{D_{J+1}}, c_2 + D_{J+1}$
    if $\lambda_{J+1} \leq h^2$ and $J < d$ **then**
      break
    end if
    $J \leftarrow J + 1$
  end while
  $\lambda_i \leftarrow h^2$ ▷ For $1 \leq i \leq J$.
  $\lambda_j \leftarrow \sigma\sqrt{\frac{2dh^2}{aD_j}}$ ▷ For $J + 1 \leq j \leq d$.
  return $\sqrt{\lambda}$
end procedure

To compute Λ^* we can naively use Theorem 10 and iterate through $K \leq J \leq d$ and compute a candidate solution λ'. If $\lambda' \preceq h^2\mathbf{1}$ then we know that it is valid solution. We then simply need to find the smallest J for which this is true, as given by Proposition 18. However, this will provide quadratic runtime to compute λ^* as in worst case we have to compute d candidates and for each candidate we need to perform d operations to produce λ'. While we will see that this is not the dominating order in the runtime of `CASG`, we found it empirically useful to improve this runtime to be linear. For this note the following lemma.

Algorithm 3 Get λ_{J+1}

1: **procedure** GET$\lambda_{J+1}(J, D, \sigma, h, c_1, c_2)$
2: $d \leftarrow$ Dimension of D
3: **if** $J = 0$ **then**
4: $J \leftarrow c_1 - \sqrt{D_1}$
5: $a \leftarrow \sqrt{2}(\frac{d\sigma^2}{D_1}(J\sqrt{8D_1(d+1)+J^2} + 2D_1(d+1) + J^2))^{1/2}$
6: $\lambda_1 \leftarrow \frac{2d}{aD_1}\left(\frac{a^2}{4d} + \sigma^2(d+1)\right)$
7: **else**
8: $c_1, c_2 \leftarrow \sigma\sqrt{\frac{dh^2}{2}}c_1, h^2c_2$
9: $disc \leftarrow 81c_1^2 - 12c_2^3$
10: **if** $disc < 0$ **then**
11: $\theta \leftarrow \cos^{-1}(9\frac{c_1}{\sqrt{12c_2^3}})$
12: $a \leftarrow \frac{4}{3}\cos(\theta/3)^2c_2$
13: **else**
14: $a \leftarrow \frac{\sqrt[3]{\frac{2}{3}}c_2}{\sqrt[3]{\sqrt{disc}+9c_1}} + \frac{\sqrt[3]{\sqrt{disc}+9c_1}}{\sqrt[3]{18}}$
15: **end if**
16: $\lambda_{J+1} \leftarrow \sigma\sqrt{\frac{2dh^2}{aD_{J+1}}}$
17: **end if**
18: **return** λ_{J+1}, a
19: **end procedure**

Lemma 32. *If $J \geq 1$ and $J \geq K$ then if λ' given by the necessary conditions in Theorem 10 is such that $\lambda'_{J+1} \leq h^2$ then $\lambda' \preceq h^2\mathbf{1}$.*

Proof. It follows from Proposition 19, where any solution to the equations $\lambda^{(\mathcal{W}_J)}$ is decreasing starting at the $J+1$th entry whenever $J \geq 1$. Hence, if $\lambda^{\mathcal{W}_J}_{J+1} \leq h^2$ then $\lambda^{\mathcal{W}_J} \preceq h^2$. □

The above results states that it suffices to compute λ_{J+1} to know whether a candidate solution is feasible. In our algorithm it remains to show that this can also be done in constant time.

In Algorithm 2 we iterate through each J and compute the $J+1$th entry. J here indicates the number of elements which have been set to h^2. It is initialized in line six to the number of non-positive elements in D as given by Theorem 10. Then the constants c_1, c_2 are initialized in the next line and updated in constant time in line ten. These values are used in line nine to compute the current λ_{J+1} via Algorithm 3, which is then checked in line eleven. If the current λ_{J+1} is less than h^2 the remaining entries are computed. Overall, providing a linear runtime.

3.4.3 Extending CASG

As discussed in the introduction to this section and as part of the base algorithm in Section 3.4.1, the Hadamard matrix does not exist for each dimension. The current conjecture is that it exists for all dimensions which are multiples of four, but there is no known algorithm to do so. However, for dimensions which are powers of two, Sylvester's algorithm is guaranteed to construct a Hadamard matrix. Therefore, we will devise an extension of the base algorithm of `CASG` which partitions any space into subspace which are of a dimensions given by a power of two. Besides the guaranteed algorithm for the construction of the Hadamard matrix in such dimensions, we as of now found there to be little benefit of using a more granular methodology, such as referring to all existing Hadamard matrix [KTR05]. However, we can envision more elaborate partitioning systems to also have merit.

Partition Set-up

Let $\{\mathcal{B}_i\}_{1\leq i\leq k}$ be a partition of the standard basis $\{e_i\}_{1\leq i\leq d}$ into k sets, with each $\mathcal{B}_i$ being a cell of the partition represented as a matrix by B_i. We let B be the concatenation of the cells, that is $B = [B_1 \quad B_2 \quad \dots \quad B_k]$. Since B is simply a reordering of the standard basis, the matrix is orthogonal. Using this partition, we enforce that each difference vector is spanned by one of the cells, that is there exists $1 \leq i \leq k$ and $c_{ji} \in \mathbb{R}^d$ such that $B_i c_{ji} = s_j$ for $1 \leq j \leq d$. We can now write $S = BS_B$, where the ith column of S_B represents the coefficients of the difference vector s_i in the corresponding cell. It follows that S_B is a block-diagonal matrix, with the ith block representing the coefficients of the differences vectors spanned by cell B_i. It immediately follows that, if it exists, the matrix S_B^{-1} is also a block diagonal matrix, with the ith block of S_B^{-1} being equal to $S_{B_i}^{-1}$.

Proposition 33. *Given a partition $\{B_i\}_{1\leq i\leq k}$ and a block-diagonal matrix S_B with S_{B_i} being the ith block providing the coordinates of the difference vectors in the space spanned by B_i, we have $\ell_{D,\sigma,h}(BS_B) = \sum_{i=1}^{k} \ell_{D_{B_i},\sigma,h}(S_{B_i})$ with $D_{B_i} = B_i D B_i^T$.*

Proof. If S_B is not invertible then at least one block will not be invertible and hence the two

sides of the equation are trivially equal as they equal ∞. When S_B is invertible the result follows immediately from the block structure of S_B and orthogonality of B. □

Given the above once result, once we have decided on a partition, we can then use the base algorithm of `CASG` to minimize the objective function in each cell of the partition. While the sample set under this construction will not be optimal, since we have imposed the partition, we have empirically found that carefully choosing B leads to strong gradient estimates.

Choosing the Partition

To use the base algorithm of `CASG` a Hadamard matrix has to exist for each cell. As discussed above, we choose cells that are dimensions of powers of two. We do so by constructing $\lfloor \log(d) \rfloor$ cells, where for $1 \leq i \leq k$ we add a cell of dimension 2^{i-1} for each $c_i = 1$ as given in the binary expansion of the dimension $d = \sum_{i=1}^{\lfloor \log(d) \rfloor} c_i 2^i$. It now remains to describe the subspace each cell will span.

We will base the span of each subspace on the intuition developed in Section 3.3. Recall, when there exist negative and positive diagonal elements then we would like to pair them together so that the approximation error can be removed. When only positive elements remain, then a small diagonal element in each cell is beneficial, as it will help structure the sample set to exploit low curvature directions. We also make the heuristic assumption that the cell with the largest dimension will have preference in the choice of elements. For example, if only positive elements in the diagonal matrix exist, then the cell with the largest dimension obtains the smallest diagonal element. This helps to reduce the error of the largest space the most.

Subdivision Algorithm

The method is given in Algorithm 4. The algorithm is greedy and continuously adds the smallest and largest remaining diagonal elements to a given cell. The reasoning for also adding the largest diagonal element is to keep size of positive and negative elements in a cell close to unity as given by Proposition 22. For the definite case, the intuition is that the largest remaining

diagonal element will have to be added to some cell. Since the current cell receives the minimal element, which has the highest effect given by Proposition 23, it seems reasonable to simply add the maximal element also to the current cell.

In practice, we line two and three we set up the binary expansions. Then initialize the cells and set the starting index for the cell enumeration to the cell with largest subspace spanned. Notice that the set of cell is zero indexed here to conform to the binary expansion and index update in line 18. We then initialize the indices of the smallest and largest remaining diagonal elements. We then proceed greedily. Whenever a cell is not yet full, we add the standard basis vector corresponding to the largest and smallest remaining diagonal elements. In the case when $i = 0$, that is the cell spans only one dimension, we only add the smallest diagonal element. We then update the start and end index to indicate the removal of the diagonal elements. We then update the cell index in line 19.

Algorithm 4 Subdivide Space

1: **procedure** SUBDIVIDE SPACE(D) ▷ D is sorted in increasing order
2: $d \leftarrow$ Dimension of D
3: $d^{(bin)} \leftarrow$ Binary expansion of d ▷ Least significant digit last; Zero-indexed
4: $\mathcal{B} \leftarrow$ Ordered set of cells ▷ Zero-indexed
5: $i \leftarrow |d^{(bin)}| - 1$ ▷ Cell index starts with most significant digit
6: $start, end \leftarrow 1, d$
7: **while** $start \leq end$ **do**
8: **if** $d_i^{(bin)} = 1$ and $|\mathcal{B}_i| < 2^i$ **then**
9: **if** $i = 0$ **then**
10: $\mathcal{B}_i = \mathcal{B}_i \cup \{e_{start}\}$
11: $start \leftarrow start + 1$
12: **else**
13: $\mathcal{B}_i = \mathcal{B}_i \cup \{e_{start}, e_{end}\}$
14: $start \leftarrow start + 1$
15: $end \leftarrow end - 1$
16: **end if**
17: **end if**
18: $i \leftarrow i - 1 \bmod |d^{(bin)}|$
19: **end while**
20: **return** $\mathcal{B}$
21: **end procedure**

3.4.4 Runtime

The runtime of `CASG` is dominated by the eigendecomposition of H, which has runtime $\mathcal{O}(d^3)$. The computation of Λ^* is done in linear time by Proposition 10 which we have found to improve the wall-clock time compared to using a naive quadratic algorithm. The Hadamard matrix is computed via the Sylvester's Construction which has runtime $\mathcal{O}(d^2)$. Lastly, the subdivision method in Algorithm 4 as presented is a quadratic time algorithm, however in practice we add the index rather than the full standard basis vector making it a linear time algorithm and improving wall-clock time. Overall, the method is a $\mathcal{O}(d^3)$ due to the eigendecomposition of the given matrix H.

While the runtime of `CASG` is larger than the linear runtime of many finite difference methods, we note that for numerical differentiation the assumption is that function evaluations are expensive. Hence, we trade computational speed for accuracy in gradient estimates and fewer function evaluations.

3.5 Global Model Framework and Numerical Results

3.5.1 Global Model Framework

We now discuss the global model framework for obtaining the Hessian needed for computing CASG. The idea behind the framework is simple. In many applications, the function will be evaluated multiple times in various regions of the domain. While the information we obtain may not be accurate enough for a gradient estimate, it may still be valuable to guide us in selecting the right sample set for a local gradient estimate. In our framework we formalize this via Algorithm 5. For the global model framework several input variables are needed. One is the evaluation history set $\mathcal{E}$ as described in Section 2.1.2. Then we need a model space and a model fitting function. We let the fitting function be given by $\Phi(\mathcal{E}) \to \phi$ which maps evaluation history set to a global model $\phi : \mathbb{R}^d \to \mathbb{R}$. In many applications we would not like to use the complete evaluation history, as some points may be poorly spread out for the chosen model space and

fitting function. Hence, we allow for a filter function on the evaluation history $\mathcal{F}(\mathcal{E}) \to \mathcal{E}_{\mathcal{F}}$ where $\mathcal{E}_{\mathcal{F}} \subseteq \mathcal{E}$. Lastly, we denote by $\mathcal{G}$ the gradient estimator which maps a Hessian estimate and the current evaluation point to a gradient estimate and also returns the function evaluations which happened as a consequence of estimating the gradient. We return the gradient estimate and the new function evaluation history.

Algorithm 5 Global Model Framework

procedure GLOBAL MODEL FRAMEWORK($\mathcal{E}, \Phi, \mathcal{F}, \mathcal{G}, x_0$)
  $\mathcal{E}_{\mathcal{F}} \leftarrow \mathcal{F}(\mathcal{E})$ ▷ Filter evaluation history
  $\phi \leftarrow \Phi(\mathcal{E}_{\mathcal{F}})$ ▷ Fit global model
  $H_{global} = \nabla^2 \phi(x_0)$
  $g, \mathcal{E}_{curr} \leftarrow \mathcal{G}(H_{global}, x_0)$ ▷ Grad estimate
  $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{E}_{curr}$ **return**
  $g, \mathcal{E}$
end procedure

While the framework requires the choice of a model space, fitting function, and an evaluation history filter, such parameters often arise from the application domain as we will see in the following applications.

A natural question is that if the global model is good enough as a Hessian estimator, should it not also be a good gradient estimator. We will empirically demonstrate this to be not true. The intuition is that a gradient estimator is allowed to sample locally, while the global model will generally not have these local evaluations present in the evaluation history and hence be less accurate close to the evaluation point.

Global Model: Cubic Spline

For our experiments we found the cubic spline radial basis function to be a good global model. We use the implementation by `scipy` [VGO+20] and extend it to allow for the analytic computation of the gradient and Hessian once the model has been fit. When some of the points in the evaluation history are close together, and the fit needs to be regularized, we also used the smoothing parameter that is part of the fitting procedure. We will specify when we use the smoothing parameter.

Preliminary Error Analysis

We briefly observe what happens when the estimate of the Hessian is slightly off and how our constraints safe-guard against this. In the end we also discuss another regularization method which can help reduce overfitting to poor Hessian estimates. We first present the following result on how the true and estimated loss functions are related. Specifically, how our constraint h allows us to bound the difference. To simplify the results, we focus on the case when the Hessian is perturbed along the diagonal.

Lemma 34. *We have that when $D^{(true)} = D^{(est)} + \Delta$ where Δ is also diagonal that*

$$\ell_{D^{(true)}}(S^*) = \ell_{D^{(est)}}(S^*) + \frac{\sum_{i=1}^d \Delta_i \lambda_i^*}{d\lambda_{max}^*} \left(2\sum_{i=1}^d D_i^{(est)} \lambda_i^* + \sum_{i=1}^d \Delta_i \lambda_i^* \right).$$

Furthermore, the following bound on the difference of the true and estimated loss holds

$$-h^4 \frac{(\sum_{i=1}^d D_i^{(est)})^2}{d\lambda_{max}^*} \leq \ell_{D^{(true)}}(S^*) - \ell_{D^{(est)}}(S^*) \leq \frac{3h^4}{d\lambda_{max}^*} \max\left((\sum_{i=1}^d |\Delta_i|)^2, (\sum_{i=1}^d D_i^{(est)})^2 \right).$$

Proof. Since the solution S^* is feasible, recall the loss is given by

$$\ell_{D^{(true)}}(S^*) = \frac{1}{4}\|S^{-T} p_{D^{(true)}}(\mathcal{S})\|^2 + \|S^{-1}\|_F^2 + \|S^{-T}\mathbf{1}\|^2.$$

The matrix $D^{(true)}$ does not affect the noise error terms. Focusing only on the approximation error first notice that $s_i^T D^{(true)} s_i = s_i^T D^{(est)} s_i + s_i^T \Delta s_i$ for $1 \leq i \leq d$. Hence, we have that $p(\mathcal{S}^*) = (\sum_{i=1}^d D_i^{(est)} \lambda_i^* + \sum_{i=1}^d \Delta_i \lambda_i^*)\mathbf{1}$ and subsequently the approximation error decomposes as

$$\begin{aligned}
\|S^{-T} p_{D^{(true)}}(\mathcal{S}^*)\|^2 &= \|S^{-T}(p_{D^{(est)}}(\mathcal{S}^*) + p_\Delta(\mathcal{S}^*))\|^2 \\
&= \|S^{-T} p_{D^{(est)}}(\mathcal{S}^*)\|^2 + \|S^{-T} p_\Delta(\mathcal{S}^*)\|^2 + 2\langle S^{-T} p_{D^{(est)}}(\mathcal{S}^*), S^{-T} p_\Delta(\mathcal{S}^*)\rangle
\end{aligned}$$

The noise terms combined with the first term of the above expansion give the loss $\ell_{D^{(est)}}(S^*)$. We simplify the remaining terms as $\|S^{-T} p_\Delta(\mathcal{S}^*)\|^2 = \frac{1}{d\lambda_{max}^*}(\sum_{i=1}^d \Delta_i \lambda_i^*)^2$ and similarly $\langle S^{-T} p_{D^{(est)}}(\mathcal{S}^*), S^{-T} p_\Delta(\mathcal{S}^*)\rangle = \frac{1}{d\lambda_{max}^*}(\sum_{i=1}^d \Delta_i \lambda_i^*)(\sum_{i=1}^d D_i^{(est)} \lambda_i^*)$.

Now, notice the following bounds on the error added to $\ell_{D^{(est)}}$

$$\begin{aligned} -\|S^{-T}p_{D^{(est)}}(\mathcal{S}^*)\|^2 &\leq \|S^{-T}p_{\Delta}(\mathcal{S}^*)\|^2 + 2\langle S^{-T}p_{D^{(est)}}(\mathcal{S}^*), S^{-T}p_{\Delta}(\mathcal{S}^*)\rangle \\ &\leq \|S^{-T}p_{\Delta}(\mathcal{S}^*)\|(\|S^{-T}p_{\Delta}(\mathcal{S}^*)\| + 2\|S^{-T}p_{D^{(est)}}(\mathcal{S}^*)\|) \\ &\leq 3\max(\|S^{-T}p_{\Delta}(\mathcal{S}^*)\|^2, \|S^{-T}p_{D^{(est)}}(\mathcal{S}^*)\|^2). \end{aligned}$$

Lastly, notice the following bounds $\frac{(\sum_{i=1}^{d}\Delta_i\lambda_i^*)^2}{d\lambda_{max}^*} \leq h^4\frac{(\sum_{i=1}^{d}|\Delta_i|)^2}{d\lambda_{max}^*}$ and $\frac{(\sum_{i=1}^{d}D_i^{(est)}\lambda_i^*)^2}{d\lambda_{max}^*} \leq h^4\frac{(\sum_{i=1}^{d}D_i^{(est)})^2}{d\lambda_{max}^*}$. The first is immediate and the latter follows from knowing that all elements associated with non-positive diagonal elements in $D^{(est)}$ are already set to h^2 and that increasing the remaining λ_i^* elements will increase the magnitude of the sum.

□

The above result allows us to reason about the error in our estimated loss value $\ell_{D^{(est)}}(S^*)$. In the best case, we have that the true error is lower than the estimated error. However, it can also happen that error is larger, directly dependent on Δ, the difference of the true and estimated Hessian.

The main observation in the bound is the dependence on $\frac{h^4}{\lambda_{max}^*}$. We now argue that by reducing h we have control in maximal error in our estimate of the loss. First we show that we can assume that $\lambda_{max}^* = h^2$. By Theorem 10, if $\lambda_{max}^* < h^2$ and vary h then λ_{max}^* is not affected. Only when h is made small enough, such that $\lambda_{max}^* = h^2$, will the maximal eigenvalue decrease with the constraint. This allows us to reason that the bound also grows roughly as h^2. While reducing h leads to control in the error of our estimate, it may also increase the loss function itself as a worse sample set will be constructed.

Another difficulty with an estimated Hessian is that the sample set may be ill-conditioned. When the Hessian is accurate, this is not an issue as it is necessary to exploit the geometry. However, when only poor estimates are available, this will adversely affect the gradient estimates. Recall that the optimal structure for the noise error, and when no information about the function is available, is given by an orthogonal matrix as shown in Lemma 24. Hence, when the Hessian

estimate is poor, we may both not be exploiting the geometry and using a structure that increases the noise error as it is ill-conditioned. Hence, future work may consider a lower-bound on the matrix norm of the sample set also. This will may avoid overfitting, as it forces a more uniform condition number.

3.5.2 Sensitivity Analysis

Sensitivity analysis is used to understand how the response of a system depends on small changes in the input. Often this is done by computing the gradient locally [QM20], which provides an ideal application area under which to assess `CASG`. To replicate a corrupted system, we add additive Gaussian noise of which we assume to know the noise level as is common for assessing gradient estimation methods [SXXN22, BBN19, BDHH05].

We compare our method against forward (FD) and central (CD) differences and the gradient estimate given by the global model, i.e. $\nabla\phi$. For comparison, our method and FFD will be given both global model information and exact Hessian information. Forward differences will be used as given in [BCCS22]. We use CD where the difference vector lengths are set to the step-size parameter h [BFB15]. This is a fair comparison as CD has no second order error and is used as a best case comparison. For our CASG, FFD and CD, the step-size hyperparameter h is found by a hyperparameter search. The global model gradient estimate is attained by simply differentiating the global model.

To assess the goodness of the gradient estimate by our method, FFD and CD, we use the exact version of the mean squared error as given in Proposition 5. While Proposition 5 is stated only for the standard simplex gradient, it can be immediately extended for central differences. The gradient estimate from the global method is assessed by considering the squared norm of the difference to the true gradient. For the global model there is no noise component as the randomness is only due to the function evaluation history, which we assume to be given for any local gradient estimate.

The main results are given in Figures 3.2 and 3.4. For Figures 3.2a, 3.2b, 3.4a and 3.4b we

chose 100 points uniformly at random from the domain and for each gradient estimation method computed the log ratio as in [MW09] between the respective method and the given version of CASG, i.e. $\log_2(\frac{MSE_{\text{method}}}{MSE_{\text{CASG}}})$.

To make the cost of using a function evaluation history more explicit we also plot the error of each method against the number of points in $\mathcal{E}$. Specifically, for Figures 3.2c and 3.4c the global method was fitted with a variable number of points as given by the x-axis. The error-bars represent the 25th and 75th percentile of the MSE values at the 100 randomly chosen points, with the line being the median.

For the global model we use no-smoothing, as the function evaluation points will be sufficiently spread out to mitigate the effect of noise.

Ackley function

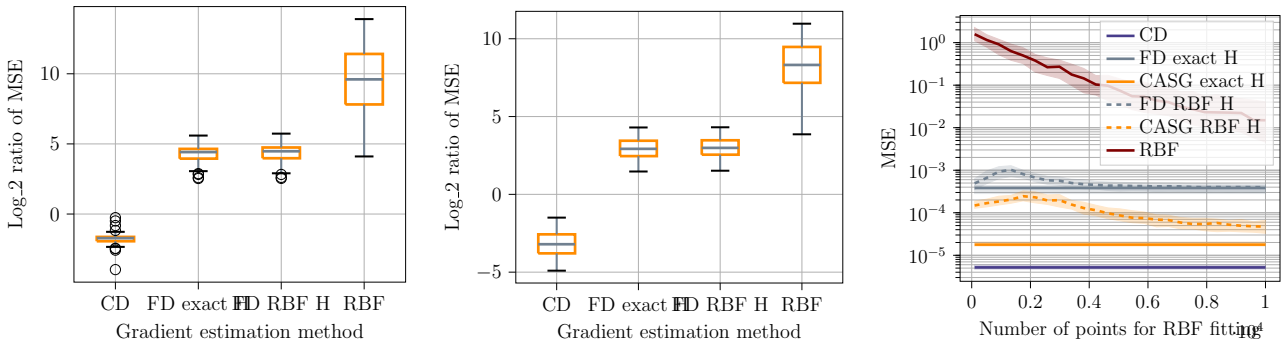
The Ackley function is a common test function due to its high degree of non-linearity and is given by $x_i \in [-0.5, 0.5]$

$$f(x_0 \cdots x_n) = -20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)) + 20 + e.$$

For the experiment we set the dimension to 8 and the noise level to $\sigma = 10^{-5}$. The set of step-sizes we consider as hyperparameter is $\{0.1, 0.05, 0.01\}$. For the RBF we sample 10^4 points for the global model for the box plots. Since the dimension is 8 we can use `CASG` directly.

ODE Model: Cell differentiation in the colon

We consider a model for cell differentiation in the colon which has been used for sensitivity analysis [QM20, JEB+07]. The process tracks three variables and is given by the following



(a) `CASG` with Exact Hessian (b) `CASG` with RBF Hessian (c) RBF points vs. MSE

Figure 3.2: Comparison of gradient estimation methods for Ackley with $d = 8$ and $\sigma = 10^{-5}$.

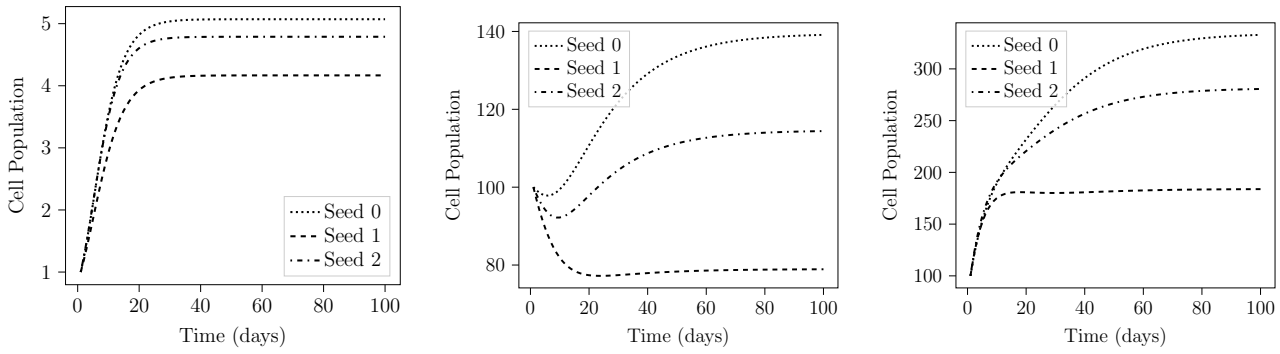
dynamics

$$\begin{aligned}
\frac{dN_0}{dt} &= (\alpha_3 - \alpha_1 - \alpha_2)N_0 - \frac{k_0 N_0^2}{1 + m_0 N_0} \\
\frac{dN_1}{dt} &= (\beta_3 - \beta_1 - \beta_2)N_1 + \alpha_2 N_0 - \frac{c_1 N_1^2}{1 + m_1 N_1} + \frac{k_0 N_0^2}{1 + m_0 N_0} \\
\frac{dN_2}{dt} &= -\gamma N_2 + \beta_2 N_1 + \frac{c_1 N_1^2}{1 + m_1 N_1}.
\end{aligned}$$

In the dynamics, the time variable t is in units of days, N_0 represents the stem cell population, N_1 the semi-differentiated cell population and N_2 the fully differentiated cell population. For the dynamics we fix the initial conditions to be $N_0 = 1$, $N_1 = 100$, $N_2 = 100$ and consider a time-span of 100 days. The remaining coefficients are the independent variables with which we perform the sensitivity analysis. Their interpretation and recommended values are given in Table 11 of [QM20]. N_0, the stem cell population, will be the dependent variable against which the sensitivity analysis is performed.

The domain we use is $\pm 10\%$ of the suggested values listed in [QM20]. To evaluate the differential equation we use the standard one-step explicit Euler scheme with a step size of 0.01, amounting to 10^4 steps per equation. To simulate noise in the observation of the final sample, we add noise with $\sigma = 10^{-3}$ which we found to be significantly more than the numerical error in simulating the dynamics. To get an understanding of how dynamics can change with different input parameters, we sample choose different parameters from the specified domain and plot them in Figure 3.3.

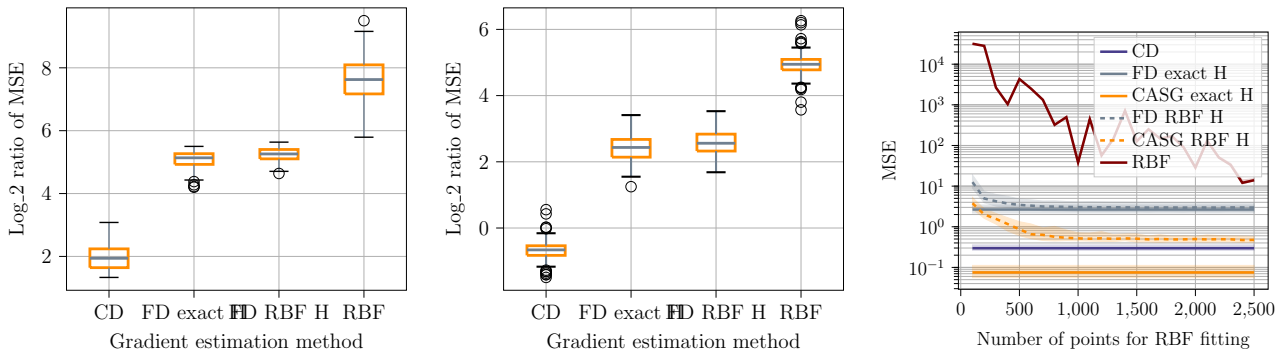
The dynamics do not have an analytic gradient easily available. Instead, we use central



(a) Stem cell population (b) Semi-Differentiated cell population (c) Fully Differentiated cell population

Figure 3.3: Visualization of ODE path with three randomly chosen parameters around the recommended values.

differences with a step-size of 10^{-11} to construct a gradient from the process without noise. We found the error due to machine precision and the discretization of the dynamics to be insignificant compared to the artificial noise added. To obtain the true Hessian we apply the same procedure to the estimated gradient values. The range of step-sizes for the gradient estimation is $\{3.16 \times 10^{-4}, 10^{-3}, 3.16 \times 10^{-3}, 10^{-2}, 3.16 \times 10^{-2}, 10^{-1}\}$. Since the number of variables is eleven, we use e`CASG` as described in Section 3.4.3. For Figure 3.4b the global model is fitted with $1,000$ points.



(a) e`CASG` using True Hessian (b) e`CASG` using RBF Hessian (c) Varying RBF Points

Figure 3.4: Comparison of gradient estimation methods for sensitivity analysis of cell differentiation in the colon with $\sigma = 10^{-3}$.

Sensitivity Analysis: Discussion

Our method outperforms the forward difference based methods and the gradient estimates by the global model by up to an order of magnitude. More interestingly, our method appears to perform close to central differences. For the ODE problem, e`CASG` with the exact Hessian even outperforms central differences.

While outperforming central differences seems like a contradiction, as it supposed to be a best-case comparison, for some problems it may be the case that the layout of the difference vectors is not only good for a second-order approximation, but also a third order approximation. Hence, it may be that the specific sample set geometry also reduces the third-order error and hence improves upon central differences. Understanding how to purposefully distribute the sample set for a central-difference type gradient estimator is left for future work. Overall it is promising that the optimal sample set can achieve a comparable error as central difference with half the number of samples.

We now note the dependence on the number of sample points. While the box-plots are given a global model with $10,000$ and $1,000$ points, this corresponds to roughly 300 and 90 gradient estimates respectively. When a good understanding of the sensitivity of the system is needed, these are numbers which can be easily achieved. Furthermore, as seen in Figure 3.2c far fewer points suffice to obtain a strong gradient estimate.

3.5.3 Derivative Free Optimization

Another application in which gradient estimation is crucial is derivative free optimization (DFO). To compare the gradient estimation methods, we use the bench-marking methodology developed in [MW09] and test on a range of CUTEst problems [GOT15]. For optimization we use a L-BFGS DFO framework from [BBN19] which allows various gradient estimation methods to be plugged in while keeping everything else fixed.

Besides the methods considered for Sensitivity Analysis, we also consider the Adaptive Finite

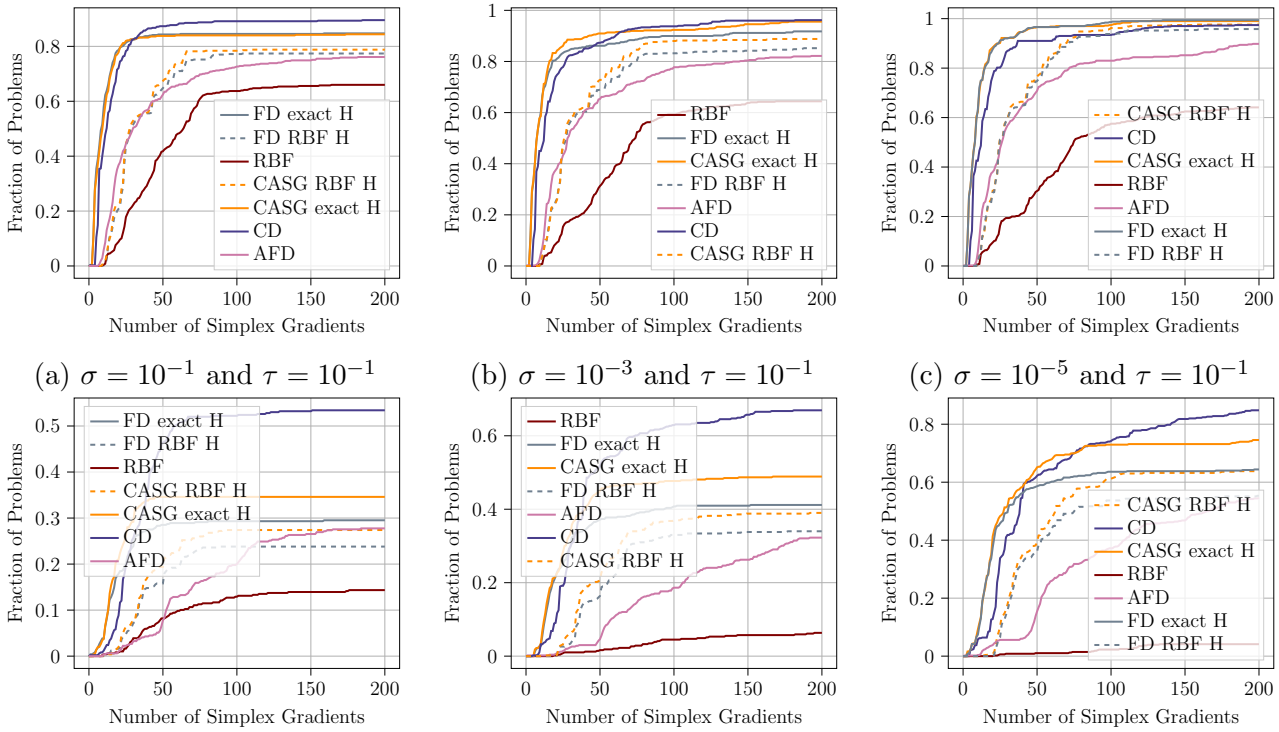
Difference (AFD) method [SXXN22]. AFD is based on forward differences and computes the lengths of difference vectors adaptively instead of using a global model. We consider AFD to be the state-of-the-art method for using finite differences in DFO as it has been shown to be more accurate and robust than comparable methods [SXXN22].

The sole hyperparameter h is set to 0.1, 0.01, and 0.001. While AFD is designed to not need such a parameter, to ensure fairness, in addition to leaving AFD with the default parameters we also assess it when upper-bounding the difference vectors by h as done for FFD and CASG. To account for noise, for each hyperparameter we obtain ten random runs. We then set the optimal hyperparameter to be the one given by the random run which achieved the lowest function value. We then have ten random-runs with the chosen hyperparameter.

To provide some initial global understanding of the function, at initialization we sample $100 \times d$ points randomly in a cube of side-length two, where d is the dimension of the specific problem. During optimization, we save the function evaluations which happen as part of gradient estimation. While other points are evaluated as part of L-BFGS, adding them to the function evaluation history made the global model perform worse. When fitting the global model, we use the minimum number of $100 \times d$ and $1,000$ data-points from the function evaluations along the optimization path. When using the the global model for gradient estimation, to obtain a function evaluation history we artificially construct a FFD sample set with difference vector lengths of 0.1 after each gradient estimation by the global model. We then add these function evaluation to $\mathcal{E}$. Lastly, the smoothing parameter is set to 0.1.

In Figure 3.5 we plot the data-profiles [MW09] of each method. To create the plots, for each CUTEst problem P we let f_L^P be the lowest average function value achieved by any gradient estimation method for problem P, with the average being taken over the corresponding random runs. We then let the convergence criterion for the CUTEst problem be $(f(x_0^P) - f_L^P) \geq (1-\tau)(f(x_0^P) - f(x_k^P)$, with x_0^P being the starting position for the problem, x_k^P the first point which satisfies the condition and τ being the tolerance. Notice that each random run has a natural convergence criterion associated via the problem it was executed on. The data-profile then plots for each gradient estimation method the fraction of random runs which achieved their

corresponding convergence criterion within a certain number of function evaluations. We use the unit of simplex gradients, which is simply the number of total function evaluations divided by the dimension of the problem.



(a) $\sigma = 10^{-1}$ and $\tau = 10^{-1}$ (b) $\sigma = 10^{-3}$ and $\tau = 10^{-1}$ (c) $\sigma = 10^{-5}$ and $\tau = 10^{-1}$

(d) $\sigma = 10^{-1}$ and $\tau = 10^{-5}$ (e) $\sigma = 10^{-3}$ and $\tau = 10^{-5}$ (f) $\sigma = 10^{-5}$ and $\tau = 10^{-5}$

Figure 3.5: Data profiles for L-BFGS using various gradient estimation methods with noise levels $10^{-1}, 10^{-3}$ and 10^{-5}. The top row are the profiles with $\tau = 10^{-1}$, associated with approximate convergence. The bottom row is $\tau = 10^{-5}$, under which the convergence criterion is only satisfied if a random run is able to get very close to the lowest achievable function value.

In Figure 3.5 the lines for methods which use the global methods are shifted by ten simplex gradient evaluations due to the initialization data. In some cases the initialization data may not be considered as part of the expense of optimization as the function may have been evaluated at various locations already. In which case the graph would be shifted over by ten simplex gradient evaluations, improving our method in terms of budget required.

DFO: Discussion.

In Figure 3.5 we observe that when $\tau = 10^{-1}$, many of the methods perform similarly, as the convergence criterion is relaxed. So, if an approximate solution is required then `CASG` and

FFD may suffice. Especially when the initialization data for the global model is not included, the methods using the global model may be favorable, as fewer evaluations than CD may be necessary.

However, when $\tau = 10^{-5}$ more accurate methods perform significantly better. When the true Hessian is used, at first both `CASG` and FFD perform better, since per gradient estimate fewer evaluations are needed. However, there comes a point where the lines intersect with CD, as then the methods can not match the accuracy of CD.

We also observe that `CASG` with the global model outperforms all other forward difference based models. Specifically, when higher accuracy is needed the gap between the two methods is significant. Additionally, even standard forward differences using our framework tends to outperform AFD, implying that our framework which uses a global model may be more suited for DFO.

We do note that AFD tends to continue improving after 200 steps. As the other methods have already long obtained convergence, we decided on a comprise to show the initial variation while showing the approximate convergence of AFD. However, AFD does not outperform `CASG` even when more evaluations are used and tends to eventually become similar to FFD using the global model framework.

Overall, `CASG` using the true Hessian far outperforms all other gradient estimation methods except for CD once enough function evaluations have been used. In practice only `CASG` with the global model framework can be used, which also outperforms all comparable methods. Importantly, `CASG` with the global model performs comparable to FFD using the true Hessian. Hence, even if a better finite difference based method is developed, `CASG` with our framework should provide an upper-bound in performance.

3.6 Proofs

3.6.1 Poofs for Theorem 9

We present here the proof for the three core Propositions which make up the proof of Theorem 9—structured into three subsections. The general layout involves proving any relevant technical Lemma, concluding with the statement of the original Proposition and the respective proof.

Proving Proposition 11: Proving the lower-bound

Lemma 35. *Let $c \in \mathbb{R}_+$, then the solution w^* to*

$$\begin{aligned} \min_{w \in \mathbb{R}^d} \quad & \|w\|_2^2 \\ s.t. \quad & \sum_{i=1}^{d} w_i = c \end{aligned}$$

is given by $w_i^* = \frac{c}{d}$ for $1 \leq i \leq d$.

Proof. The proof is a simple application of Lagrange multipliers. □

Lemma 36. *Let V be an orthogonal matrix and D a diagonal matrix. Then for any orthogonal matrix U and diagonal matrix Λ we have if $S = U\Lambda V^T$*

$$\|p_D(\mathcal{S})\|^2 \geq \frac{1}{d} tr(\Lambda U^T D U \Lambda)^2.$$

$$\|[v_i^T \Lambda U^T D U \Lambda v_i]_{1 \leq i \leq d}^T\|_2^2 \geq \frac{1}{d} tr(\Lambda U^T D U \Lambda)^2.$$

Proof. Due to the cyclic property of the trace we have

$$\sum_{i=1}^{d} v_i^T \Lambda U^T D U \Lambda v_i = \mathrm{tr}(V \Lambda U^T D U \Lambda V^T)$$
$$= \mathrm{tr}(\Lambda U^T D U \Lambda).$$

By Lemma 35 the vector $w^* \in \mathbb{R}^d$ which minimizes $\|w\|_2^2$ under the constraint $\sum_{i=1}^d w_i = \mathrm{tr}(\Lambda U^T D U \Lambda)$ is given by $w_i^* = \frac{1}{d}\mathrm{tr}(\Lambda U^T D U \Lambda)$. □

Proposition 37 (Restated Proposition 11)**.** *For a diagonal matrix $D \in \mathbb{R}^{d\times d}$ and positive real numbers $\sigma, h \in \mathbb{R}_+$ define the function*

$$\underline{\ell}_{D,\sigma,h}(U,\Lambda) = \begin{cases} \frac{a_{D,\sigma}(U,\Lambda)^2}{4d\lambda_{max}} + \sigma^2 \sum_{i=1}^d \frac{1}{\lambda_i} + \sigma^2 \frac{d}{\lambda_{max}} & \text{if } hI \succeq \Lambda \succ 0 \\ \infty & \text{otherwise,} \end{cases} \tag{3.6.1}$$

with orthogonal $U \in \mathbb{R}^{d\times d}$, diagonal $\Lambda \in \mathbb{R}^{d\times d}$, and $a_{D,\sigma}(U,\Lambda) = tr(\Lambda U^T D U \Lambda)$. Then we have

$$\underline{\ell}_{D,\sigma,h}(U,\Lambda) \leq \min_V \ell_{D,\sigma,h}(U,\Lambda,V).$$

Proof. Consider the case when $hI \succeq \Lambda \succ 0$. Otherwise, the functions both evaluate to ∞ and the bound trivially holds on the extended real number line.

Recall that $\ell(U,\Lambda,V) = \frac{1}{4}\|S^{-T}p_D(\mathcal{S})\|_2^2 + \sigma^2\|S^{-1}\|_F^2 + \sigma^2\|S^{-1}\mathbf{1}\|_2^2$. By standard norm inequalities we have that $\|S^{-T}p_D(\mathcal{S})\|_2^2 \geq \frac{1}{\lambda_{max}}\|p_D(\mathcal{S})\|_2^2$ and $\|S^{-T}\mathbf{1}\|^2 \geq \frac{d}{\lambda_{max}}$. And from the definition of the Frobenius norm we have $\|S^{-1}\|_F^2 = \sum_{i=1}^d \frac{1}{\lambda_i}$. Combining these we have that

$$\ell(U,\Lambda,V) \geq \frac{1}{4}\frac{1}{\lambda_{max}}\|p_D(\mathcal{S})\|_2^2 + \sigma^2 \sum_{i=1}^d \frac{1}{\lambda_i} + \sigma^2 \frac{d}{\lambda_{max}}.$$

By taking the minimum over V on both sides completes the proof. □

Proving Proposition 12: Finding solutions to the lower bound

In the previous section we established a lower-bound to the objective function l which does not depend on the right singular vectors V. In this Section we derive the solutions to this lower-bound. To do so, we need a couple technical lemmas. Lemma 38 focuses on deriving the form of all stationary matrices U for a given diagonal and positive definite Λ. While there are several forms, due to a natural ordering and Lemma 39 we can reduce the number of solutions and conclude with the proof of Proposition 12.

We now present the first Lemma.

Lemma 38. *Fix a diagonal matrix $\Lambda \succ 0$ and we have that*

- *the stationary points of $a(\cdot, \Lambda)$ are characterized by the set of permutation matrices and,*
- *the stationary points of $a(\cdot, \Lambda)^2$ are characterized by the set of permutation matrices and all matrices U_0 such that $a(U_0, \Lambda) = 0$.*

Proof. The Lagrangian for $a(U, \Lambda)$ is given by $\mathcal{L}(U, \mu) = tr\Big(\Lambda U^T D U \Lambda\Big) - \sum_{i,j} \mu_{ij}(u_i^T u_j - \delta_{ij})$. Taking the gradient with respect to U and setting to zero we obtain

$$2U^T D V \Lambda^2 = (\mu + \mu^T).$$

Due to $\mu + \mu^T$ we have the symmetry relation

$$U^T D U \Lambda^2 = \Lambda^2 U^T D V$$

which means that $U^T DU$ and Λ^2 commute, implying that they are simultaneously diagonalizable. Therefore, Λ^2 and $U^T DU$ share the same eigenvectors.

Since Λ^2 is diagonal, it has the standard basis vectors as eigenvectors. The set of orthogonal matrices that represent the standard basis are permutation matrices P. Hence, $U^T DU$ can be diagonalized in the basis of P which results in $U^T DU = PDP^T$. This implies that $U^T DU$

is equivalent to permuting the diagonal matrix D according to P. While technically U itself does not have to be a permutation matrix, its action on $a(\cdot, \Lambda)$ is equivalent to the action of the permutation matrix P, i.e. $a(U, \Lambda) = a(P, \Lambda)$. This allows us to characterize all stationary points of $a(\cdot, \Lambda)$ as permutation matrices.

For minimizing $a^2(U, \Lambda)$ we have that by setting the gradient of the associated Lagrangian to zero provides

$$4tr\Big(\Lambda U^T DU\Lambda\Big)U^T DU\Lambda^2 = (\mu + \mu^T).$$

If it exists, the equation can be satisfied by an orthogonal matrix U_0 such that $tr\Big(\Lambda U^T DU\Lambda\Big) = a(U_0, \Lambda) = 0$. Otherwise, just as before, we see that $U^T DU$ and Λ^2 commute, implying that U can be characterized by a permutation matrix. □

Remark 5. *Letting $U_{max}(\Lambda)$ and $U_{min}(\Lambda)$ be the permutation matrix which respectively maximize and minimize $a(U, \Lambda)$ for a given Λ, if there exists U_0 such that $a(U_0, \Lambda) = 0$ it immediately follows that*

$$a(U_{max}(\Lambda), \Lambda) \geq 0 \geq a(U_{min}(\Lambda), \Lambda).$$

By showing that we can discard U_0, which means that we only have to work with permutation matrices, we can significantly simplify the problem. To show this, we first present the following result which tells us that for the optimal $(\underline{U}^*, \underline{\Lambda}^*)$ the sign of $a(\underline{\Lambda}^*, U_{ext})$ is the same for all extrema U_{ext}.

Lemma 39. *Without loss of generality assume that $\sum_{i=1}^d D_i \geq 0$ and let $(\underline{U}^*, \underline{\Lambda}^*)$ be optimal. Then for all extrema U_{ext}*

- *if $\sum_{i=1}^d D_i = 0$ then $a(\underline{\Lambda}^*, U_{ext}) = 0$ and $\underline{\Lambda}^* = hI$ and $\underline{U}^* = I$;*
- *if $\sum_{i=1}^d D_i > 0$ then $a(\underline{\Lambda}^*, U_{ext}) \geq 0$.*

Proof. We proceed by cases.

(i) Letting $\sum_{i=1}^{d} D_i = 0$ we have that for any Λ and all extrema U_{ext}

$$\frac{1}{4\lambda_{max}}\frac{a(U_{ext},\Lambda)}{d} + \sigma^2\sum_{i=1}^{d}\frac{1}{\lambda_i} + \sigma^2\frac{1}{\lambda_{max}}d \geq \sigma^2\sum_{i=1}^{d}\frac{1}{\lambda_i} + \sigma^2\frac{1}{\lambda_{max}}d.$$

To minimize $\sigma^2\sum_{i=1}^{d}\frac{1}{\lambda_i} + \sigma^2\frac{1}{\lambda_{max}}d$ we want to maximize each element along the diagonal Λ, which is achieved when $\Lambda = hI$. Hence, $\underline{\Lambda}^* = hI$ as this will minimize the loss for all extrema U_{ext} chosen. Specifically, I is a permutation matrix and hence $\underline{U}^* = I$.

(ii) Assume that $\sum_{i=1}^{d} D_i > 0$. If $D \succeq 0$ the result is immediate. Hence, assume that at least one entry of D is negative. For contradiction, assume that for $\underline{\Lambda}^*$ we have $a(\underline{\Lambda}^*, U_{min}) < 0$, where U_{min} is the permutation matrix which minimizes $a(\underline{\Lambda}^*, U)$. For ease of notation and without loss of generality, we assume that $\underline{\Lambda}^*$ is sorted such that $U_{min} = I$. We proceed by showing that we can find a $\Lambda \succeq \underline{\Lambda}^*$ which results in a reduction of the objective function leading to a contradiction.

For the proof we introduce some notation. Recall K is the number of non-positive elements and since D is increasing we have $D_i \leq 0$ for $1 \leq i \leq K$. Let $c = (\sum_{i=1}^{K}|D_i|\underline{\lambda}_i^*)/(\sum_{i=1}^{K}|D_i|)$ where $\sqrt{c} \leq h$ since by assumption $\underline{\Lambda}^* \preceq hI$. Lastly, define a function $\lambda(\alpha) : [0,1] \to \mathbb{R}^{d\times d}$ such that

$$\lambda_i(\alpha) = \begin{cases} \lambda_i + \alpha(c - \lambda_i) & \text{if } \lambda_i \leq c, i \geq K+1 \\ \lambda_i & \text{otherwise.} \end{cases}$$

We show that given this construction $a(\underline{\Lambda}^*(1), U_{min}) \geq 0 > a(\underline{\Lambda}^*(0), U_{min})$. The lower bound is immediate by assumption. Letting $\alpha = 1$

$$\begin{aligned}\sum_{i=K+1}^{d} D_i\underline{\lambda}_i^*(1) &\geq c\sum_{i=K+1}^{d} D_i \\ &\geq \sum_{i=1}^{K}|D_i|\underline{\lambda}_i^*.\end{aligned}$$

The last step follows from the definition of c and $\sum_{i=1}^{K}|D_i| < \sum_{i=K+1}^{d} D_i$ due to the assumption $\sum_{i=1}^{d} D_i > 0$. By the above $\sum_{i=K+1}^{d} D_i\underline{\lambda}_i^*(1) - \sum_{i=1}^{K}|D_i|\underline{\lambda}_i^* \geq 0$ which by definition implies

$a(\underline{\Lambda}^*(1), U_{min}) \geq 0$.

By the above and continuity of $a(\Lambda(\cdot), U_{min}))$, we have that due to the intermediate value theorem there exists an $\alpha_0 \in (0, 1]$ such that $a(\underline{\Lambda}^*(\alpha_0), U_{min}) = 0$. Combined with the fact that $\underline{\Lambda}^*(\alpha_0) \succeq \underline{\Lambda}^*$, and hence noise error has not been increased, the objective function has been strictly decreased by using $\underline{\Lambda}^*(\alpha_0)$ leading to a contraction. □

Proposition 40 (Restated Proposition 12)**.** *The global minimizers $\underline{U}^*$ and $\underline{\Lambda}^*$ of $\underline{\ell}_{D,\sigma,h}(U, \Lambda)$ are given by $\underline{U}^*$ being the identity matrix and $\underline{\Lambda}^*$ being a diagonal matrix given by $\min_\Lambda \underline{\ell}_{D,\sigma,h}(I, \Lambda)$.*

Proof of Proposition 12. If $\sum_i D_i = 0$ we are immediately done as by the first part of Lemma 39 $U = I$. When $\sum_i D_i > 0$, we have that $a(\underline{\Lambda}^*, \underline{U}^*) \geq 0$ which implies that even if U_0 exists

$$a(U_{max}(\Lambda), \Lambda) \geq a(U_0(\Lambda), \Lambda) \geq a(U_{min}(\Lambda), \Lambda) \geq 0$$

and hence U_{min} is optimal. If U_0 does not exist, then trivially $\underline{U}^* = U_{min}$. If U_{min} is not the identity matrix, we can permute U_{min} and any Λ without changing the objective function. Hence, $\underline{U}^* = I$ and $\underline{\Lambda}^* = \arg\min_\Lambda \underline{\ell}(I, \Lambda)$. □

Proving Proposition 13: Making the objective function and the lower-bound equal

Proposition 41 (Restated Proposition 13)**.** *Let $\Lambda \succ 0$ be diagonal and M_k be the Hadamard matrix with $k = \arg\max_i \Lambda_i$. We then have that $\underline{\ell}_{D,\sigma,h}(I, \Lambda) = \ell_{D,\sigma,h}(I, \Lambda, M_k)$.*

Proof. We proceed by showing that the terms in the respective objective functions $l(I, \Lambda, M_k)$ and $\underline{\ell}(I, \Lambda)$ can be matched up and shown to be equal. For this, let $S = \Lambda(M_k)^T$.

To show that the first term of the objective functions are equal note that

$$\begin{aligned} s_i^T D s_i &= \sum_{j=1}^{d} ((M_k)_{i,j})^2 D_j (\Lambda_j)^2 \\ &= \frac{1}{d} \sum_{j=1}^{d} D_j (\Lambda_j)^2 && \text{Due to } M_k \text{ being a Hadamard matrix.} \\ &= \frac{1}{d} \text{tr}(\Lambda^T D \Lambda) = \frac{1}{d} a(I, \Lambda). \end{aligned}$$

Hence, we have that the vectorized quadratic form is constant, specifically $p_D(\mathcal{S}) = \frac{1}{d} a(I, \Lambda)\mathbf{1}$. Now, recall that $M_k \mathbf{1} = e_k$, by construction of the k-th row being all ones and M_k being orthogonal. Therefore, we have that $\|S^{-T} p_D(\mathcal{S})\|_2^2 = \frac{a(I,\lambda)^2}{d^2} \|\Lambda^{-1} M_k \mathbf{1}\|_2^2 = \frac{a(I,\lambda)^2}{d\lambda_k^2}$ and hence the first term of the objective function l matches the corresponding term of the lower-bound $\underline{\ell}$.

The second terms are trivially equal since the Frobenius norm is the sum of the squares of singular values. The third terms are equivalent due to a similar argument as the first term. Here we have that $\|S^{-T}\mathbf{1}\|^2 = \|\Lambda^{-1} M_k \mathbf{1}\|^2 = \frac{d}{\lambda_k}$ completing the result. □

3.6.2 Proofs for Theorem 10

We now present the proofs for Propositions 19 and 20. We first present the equations that any solution has to satisfy.

Proposition 42 (Restated Proposition 19)**.** *Let $K \leq J \leq d$ then for $\ell^{(proxy)}(\lambda^{(\mathcal{W}_J)}) = 0$ it is necessary for $\lambda^{(\mathcal{W}_J)}$ to satisfy the following equations*

$$\lambda_1^{(\mathcal{W}_J)} = \begin{cases} h^2 & \text{if } J = 0, \\ \frac{2d}{aD_1} \left(\frac{a^2}{4d} + \sigma^2 (d+1) \right) & \text{otherwise;} \end{cases}$$

$$\lambda_i^{(\mathcal{W}_J)} = \begin{cases} h^2 & \text{if } i \leq J, \\ \sigma \sqrt{\frac{2d\lambda_1^{(\mathcal{W}_J)}}{aD_j}} & \text{otherwise,} \end{cases} \qquad \text{for } 2 \leq i \leq d,$$

$$a = \sum_{i=J+1}^{d} D_i \lambda_i^{(\mathcal{W}_J)} + h^2 \sum_{i=1}^{J} D_i.$$

The proof of this proposition follows immediately from setting the gradient of the proxy function to zero and rearranging terms.

Proof. Let $a = \sum_{i=J+1}^{d} D_i \lambda_i^{(\mathcal{W})} + h^2 \sum_{i=1}^{J} D_i$ and $\mathbb{1}$ be the indicator function. Then the stationary points are simply all the points which satisfy

$$\frac{\partial \ell^{(\text{proxy})}(\lambda^{(\mathcal{W})})}{\partial \lambda_1} = \mathbb{1}(1 \notin \mathcal{W}_J) \left(\frac{D_1}{2d\lambda_1^{(\mathcal{W}_J)}} a - \frac{1}{4d(\lambda_1^{(\mathcal{W}_J)})^2} a^2 - \sigma^2 \frac{d+1}{(\lambda_1^{(\mathcal{W}_J)})^2} \right) = 0 \tag{3.6.2}$$

$$\frac{\partial \ell^{(\text{proxy})}(\lambda^{(\mathcal{W})})}{\partial \lambda_i} = \mathbb{1}(i \notin \mathcal{W}_J) \left(\frac{D_i}{2d\lambda_1^{(\mathcal{W}_J)}} a - \sigma^2 \frac{1}{\lambda_i^{(\mathcal{W}_J)}} \right) = 0. \tag{3.6.3}$$

To manipulate the above equations we note that if not all constraints are set, that is $J < d$, it holds that $a > 0$ for every stationary point. This is because a is non-negative due to Lemma 39 and $a \neq 0$ as otherwise the stationary point can not exist since Equation 3.6.3 can not be satisfied. It also holds that $D_i > 0$ for all $J + 1 \leq i \leq d$ since $K \leq J$. The result then follows by simple algebra. □

To find a valid λ to satisfy the above equations, it suffices to find the solution to a. The following result provides this. Note that if a current working set $\mathcal{W}_J$ does not provide a candidate solution, then the a given below will make some of the equations infeasible. Specifically, the predicted $\lambda^{(\mathcal{W}_J)} \succ h^2 \mathbf{1}$ which is not possible.

Proposition 43 (Restated Proposition 20)**.** *Let $K \leq J \leq d$, then for any candidate solution it follows that a is given by*

- *If $J = 0$, then we have*

$$a = \sqrt{2} \left(\frac{d\sigma^2}{D_1} \left(C\sqrt{8D_1(d+1) + C^2} + 2D_1(d+1) + C^2 \right) \right)^{1/2}$$

with $C = \sum_{i=2}^{d} \sqrt{D_i}$;

- *Otherwise,*

$$a = \begin{cases} \frac{\sqrt[3]{\frac{2}{3}}c_2}{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}} + \frac{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}}{\sqrt[3]{18}} & \text{if } 27c_1^2 - 4c_2^3 \geq 0, \\ \cos(\theta/3)2\sqrt{\frac{c_2}{3}} & \text{otherwise,} \end{cases}$$

where $c_1 = \sigma\sqrt{\frac{dh^2}{2}}\sum_{i=J+1}^{d}\sqrt{D_i}$, $c_2 = \sum_{i=1}^{J} D_i h^2$ *and* $\theta = arccos(\frac{9c_1}{\sqrt{12c_2^3}})$.

The following Lemmas prove Proposition 20 by splitting it into two cases.

Lemma 44. *When* $J = 0$ *then the unique non-negative solution to* a *as given by the equations in Proposition 19 is*

$$a = \sqrt{2}\left(\frac{d\sigma^2}{D_1}\left(C\sqrt{8D_1(d+1)+C^2} + 2D_1(d+1) + C^2\right)\right)^{1/2}$$

with $C = \sum_{i=2}^{d}\sqrt{D_i}$.

Proof. Let λ be given by the equations in Proposition 19. We rewrite $a = \sum_{i=1}^{d} D_i\lambda_i$ by plugging in λ_i to obtain

$$a = \sigma\sqrt{\frac{2d\lambda_1}{a}}\sum_{i=2}^{d}\sqrt{D_i} + \lambda_1 D_1.$$

Letting $C = \sum_{i=2}^{d}\sqrt{D_i}$, substituting λ_1 and simplifying we get

$$a = \frac{1}{a}\left(\sigma\sqrt{\frac{2d}{D_1}\left(\frac{a^2}{2} + 2\sigma^2 d(d+1)\right)}C + \left(\frac{a^2}{2} + 2\sigma^2 d(d+1)\right)\right).$$

The equation can be rewritten as solving for the roots of a fourth-order polynomial

$$\left(\frac{a^2}{2} - 2\sigma^2 d(d+1)\right)^2 - C^2\frac{2d\sigma^2}{D_1}\left(\frac{a^2}{2} + 2\sigma^2 d(d+1)\right) = 0.$$

However, during the algebraic manipulation, additional roots may be have been introduced.

The potentially complex roots of the fourth order polynomial are given by two pairs

$$a_1' = \pm\sqrt{2}\left(\frac{d\sigma^2}{D_1}\left(C\sqrt{8D_1(d+1)+C^2}+2D_1(d+1)+C^2\right)\right)^{1/2}$$
$$a_2' = \pm\sqrt{2}\left(\frac{d\sigma^2}{D_1}\left(-C\sqrt{8D_1(d+1)+C^2}+2D_1(d+1)+C^2\right)\right)^{1/2}$$

which can be easily verified by plugging into the polynomial.

It is easy to verify that while a_1' satisfies the original equation, a_2' will not. Of the pair a_1', by assumption only the non-negative can be valid. Since all terms under square root are positive, the solution is also real and hence provides the unique solution

$$a = \sqrt{2}\left(\frac{d\sigma^2}{D_1}\left(C\sqrt{8D_1(d+1)+C^2}+2D_1(d+1)+C^2\right)\right)^{1/2}.$$

□

Lemma 45. *When $J \geq 1$ the solution for a given by the equations in Proposition 19 is given by*

$$a = \begin{cases} \frac{\sqrt[3]{\frac{2}{3}}c_2}{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}} + \frac{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}}{\sqrt[3]{18}} & \text{if } 27c^1 - 4c_2^3 \geq 0, \\ \cos(\theta/3)2\sqrt{\frac{c_2}{3}} & \text{otherwise,} \end{cases}$$

where $c_1 = \sigma\sqrt{\frac{2dh^2}{a}}\sum_{i=J+1}^{d}\sqrt{D_i}$ and $c_2 = \sum_{i=1}^{J} D_i h^2$.

For the proof we note the following Lemma

Lemma 46. *For $c_1 > 0$ and $c_2 \in \mathbb{R}$ there exists only one positive real root for the polynomial $f(x) = x^3 - c_2 x - c_1$.*

Proof. Since $c_1 > 0$ and for any polynomial the negative of the order zero coefficient is the product of the roots, the product of the roots is positive and no root is zero. Furthermore, the second order term presents the sum of the roots, which is zero. For the sum of the roots to be zero, at least one root has to have a negative real component. For the product to be positive

there has to exist another root with negative real component and one positive real root. Hence, there exists a unique real positive root. □

Poof of Lemma 45. Again, let λ be given by the equations in Proposition 19. Plugging the equations for λ_i into $a = \sum_{i=1}^{d} D_i\lambda_i$ and after some simplification we get

$$a^{3/2} = \sigma\sqrt{2dh^2}\sum_{i=J+1}^{d}\sqrt{D_i} + \sqrt{a}\sum_{i=1}^{J} D_i h^2.$$

Letting $c_1 = \sigma\sqrt{\frac{2dh^2}{a}}\sum_{i=J+1}^{d}\sqrt{D_i}$, $c_2 = \sum_{i=0}^{d} D_i h^2$, and $x = \sqrt{a}$ the equation becomes

$$x^3 = c_1 + c_2 x$$

when restricting $x \geq 0$. By Lemma 46 and $c_1 > 0$, there exists a unique positive x satisfying the equation which provides the solution $\sqrt{a}$.

We assert the root is given by

$$x = \frac{\sqrt[3]{\frac{2}{3}}c_2}{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2 - 4c_2^3} + 9c_1}} + \frac{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2 - 4c_2^3} + 9c_1}}{\sqrt[3]{18}}.$$

We now need show that the root is always well-defined, real, and non-negative. The root can be easily verified to be a root by plugging it into the polynomial.

To show well defined, real, and non-negative we split in two cases: $27c_1^2 - 4c_2^3 \geq 0$ and $27c_1^2 - 4c_2^3 < 0$.

(i) Assume that $27c_1 - 4c_2^3 \geq 0$. It is well defined because then $\sqrt[3]{\sqrt{3}\sqrt{27c_1^2 - 4c_2^3} + 9c_1} > 0$ since $c_1 > 0$. Furthermore, this quantity is real since every individual term is real. We now need to show non-negativeness. If $c_2 \geq 0$ it is trivial, hence assume $c_2 < 0$ for which we have

$$\frac{\sqrt[3]{\frac{2}{3}}c_2}{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}}+\frac{\sqrt[3]{\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1}}{\sqrt[3]{18}}$$
$$>-\frac{\sqrt[3]{\frac{2}{3}}|c_2|}{\sqrt[3]{\sqrt{3}\sqrt{4|c_2|^3}}}+\frac{\sqrt[3]{\sqrt{3}\sqrt{4|c_2|^3}}}{\sqrt[3]{18}}=0.$$

(ii) $27c_1-4c_2^3<0$ implies $c_2>0$ since $c_1>0$, which yields

$$\begin{aligned}\sqrt{3}\sqrt{27c_1^2-4c_2^3}+9c_1 &= i\sqrt{3}\sqrt{4c_2^3-27c_1^2}+9c_1\\ &= r\left(\cos(\theta)+i\sin(\theta)\right)\\ &= re^{i\theta}=z\end{aligned}$$

where $\theta=\arccos(\frac{9c_1}{r})$ and after simplifying $r=\sqrt{12c_2^3}$. Since $c_2>0$ we have $r>0$ and hence the root is well-defined. Substituting and simplifying yields that

$$x=\cos(\theta/3)2\sqrt{\frac{c_2}{3}}.$$

which shows that the root is real. The root is non-negative because $\cos(\theta)=\frac{9c_1}{r}>0$ and hence $\theta\in(-\frac{\pi}{2},\frac{\pi}{2})$, which implies that $\theta/3\in(-\frac{\pi}{2},\frac{\pi}{2})$ and $\cos(\theta/3)>0$. □

Proof of Proposition 20. The result follows from combining Lemma 44 and 45. □

3.7 Summary

In this chapter we addressed the optimal sample set for the fully-determined simplex gradient. As claimed in our contributions, in Theorem 9 we provided the first optimality result for this simplex gradient which does not impose any constraints on the sample set. Based on our results, we introduce `CASG` and `eCASG` as sample set construction methods. We then evaluated our methods via the historically informed global model framework on sensitivity analysis and

derivative free optimization applications. We demonstrated an order of magnitude improvement over current methods. Importantly, in all sensitivity analysis experiments we are *comparable to central differences, the de facto gradient estimator, for half the cost*. With the above results, we were able to dispel the commonly held notion that the optimal sample set structure is orthogonal. In fact, we found that the structure is aimed at exploiting *directions of low curvature*. A crucial insight into optimal sample set structures when using few sample points, as it allows for more principled development of gradient estimators aimed at minimizing cost.

Chapter 4

Optimal Sample Sets Selection for the Centered Simplex Gradient

In this chapter we study the centered simplex gradient. For this setting, we assume that $2d$ function evaluations can be performed, which is in contrast to the setting of Chapter 3 where only half the budget was available. The benefit of using more function evaluations is that a symmetric structure can be imposed, which immediately removes the approximation error, which was the central study of the previous chapter. So far, no systematic study of this sample set has been performed, a gap we aim to address.

To study the centered simplex gradient we proceed as follows. As in Chapter 3, we describe the optimization problem in Section 4.2. We also provide an extensive case study in Section 4.2.4 to further motivate and describe this optimization problem. This study provides insight into the best-case performance of any centered simplex gradient estimator and is used to assess our following results. In Section 4.3 we make the link between the optimal sample set and a set of orthogonal roots of a third order homogeneous polynomial. In the same section, we immediately provide the result via Corollary 73 that such a sample set does not exist for every polynomial, and hence for every gradient estimation problem. In Section 4.4.2 we then provide Theorem 75 which asserts that $d-1$ orthogonal roots do exist for any homogeneous polynomial of odd degree. The negative result from Corollary 73 then makes this result tight. In the same

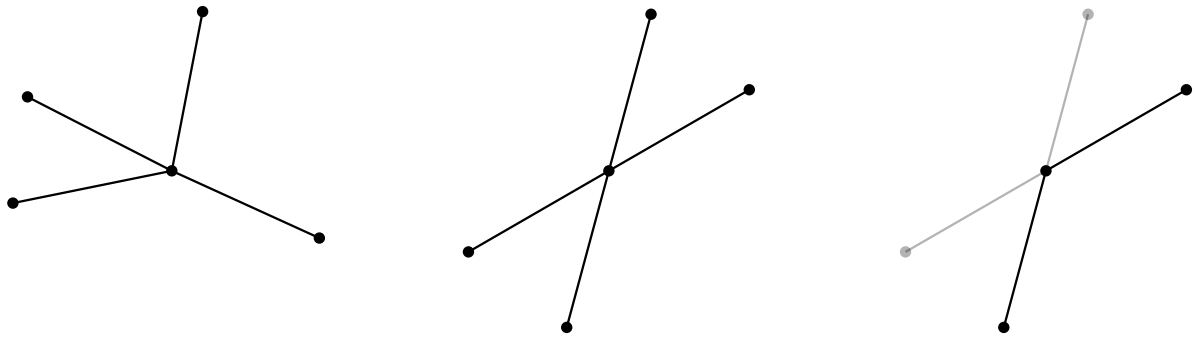
section, we describe ORCAS , a complete methodology based on Theorem 75. We then provide worst-case guarantees on our method which show that with increasing dimension the performance approaches the best-case performance and provide concrete bounds on the improvement over standard central difference methods. In Section 4.5 we discuss the impracticality of naive numerical optimal sample set solvers. Subsequently, in Section 4.5.2 we discuss how to use ORCAS as a dimensionality reduction method to be used in conjunction with a numerical solver for the sample set. This combination allows us to circumvent the orthogonality constraints imposed on ORCAS. We conclude this chapter with numerical results in Sections 4.6 and 4.7. Here, we demonstrate both theoretical and practical performance of our methods.

4.1 Background

Although the problem definition and basic setup have been presented in Chapter 2, here we present results specific to the discussion of the centered simplex gradient. We first cover a reformulation of the centered simplex gradient which will significantly ease the presentation and discussion in this chapter. We then present further technical results relating to homogeneous polynomials, which will lie at the core of our main results.

4.1.1 Centered Simplex Gradient

We work with a reformulated version of the centered simplex gradient. To define this version, we first introduce the standard centered simplex gradient and then introduce the equivalent reduced version.



(a) Simplex Gradient (b) Centered Simplex Gradient (c) Reduced CSG

Figure 4.1: The structure of various simplex gradient instances and formulations. Dots representing sample points.

Centered Simplex Gradient

The centered simplex gradient is the simplex gradient given a sample set satisfying the following two conditions

1. $N = 2d + 1$ [1],
2. $s_{2i} = -s_i$ for $1 \leq i \leq d$.

When a sample set, difference set or difference matrix satisfy these conditions we call them **centered** and we denote them by an over-bar, i.e. the centered sample set is $\bar{\mathcal{X}}$.

The centered simplex gradient is then given by

$$\nabla_{\bar{S}} f(x_0) = \bar{S}^{+} \tilde{\mathbf{f}}_{\bar{\mathcal{X}}}$$

where the function difference vector was replaced with the function vector due to the symmetry of the sample set removing the requirement of evaluation $\tilde{f}$ at x_0. An instance of a centered simplex gradient is the commonly used central differences[HJB20]. However, while central differences uses orthogonal difference vectors, this is not a requirement as we see in Figure 4.1b.

[1] In practice, only $2d$ points are required, since x_0 is not needed to fit the centered simplex gradient.

As long as a sample set satisfies the above symmetry conditions it is a centered simplex gradient and obtains significantly improved error qualities. Specifically, the approximation error, the power of the simplex gradient to approximate the true gradient in the absence of noise, improves from $\mathcal{O}(R(\mathcal{S})^2)$ to $\mathcal{O}(R(\mathcal{S})^3)$. This is a significant improvement and immediately removes the error that was minimized in [LPKJ23] at the expense of adding twice the number of function evaluations.

Reduced Centered Simplex Gradient

In the centered simplex gradient formulation only the first $d+1$ elements of the sample set are needed to fully define the set. The remaining points are simply reflected around the central evaluation point. This is apparent in Figures 4.1b and 4.1c.

To simplify the study of the centered simplex gradient, we introduce the **reduced centered simplex gradient**. For this formulation, we consider sample sets with $d+1$ elements. To emphasize that they are used for the reduced simplex gradient we may call them **reduced**. The reduced sample set will be assumed to satisfy Assumption 2, as for the standard simplex gradient.

To obtain the centered sets from reduced sample sets we introduce a **centering operator** L. The operator in matrix form is given by $L \in \mathbb{R}^{d\times 2d}$ and is defined component-wise as $L_{i,j} = \delta_{ij}$ for $1 \leq i, j \leq d$ and $L_{i,j} = -\delta_{ij}$ when $1 \leq i \leq d$ and $1 \leq j \leq 2d$. In other words, we horizontally concatenate the identity matrix with the negative of the identity matrix. We indicate a quantity which was centered by a lower-case L. Here $S_L = SL = \bar{S}$ and the centered sample set is then given by $\mathcal{X}_L = \mathcal{X} \cup \{x_0 - 2s_i\}_{1\leq i\leq d} = \bar{\mathcal{X}}$.

We now list a few useful properties of the centering operator which significantly simplify future discussion.

Proposition 47. *Let $\mathcal{X}$ be a sample set with $d+1$ elements satisfying Assumption 2. We then have that*

- $LL^T = 2I$,

- $L\mathbf{1} = 0$,
- $(S_L^T)^+ = \frac{1}{2}S^{-T}L$.

Proof. The first and second are immediate. For the third property we use the first property and note that S is invertible by assumption of being full-rank and square.

$$\begin{aligned}(\bar{S}^T)^+ &= (\bar{S}\bar{S}^T)^{-1}\bar{S} \\ &= (SLL^TS^T)^{-1}SL \\ &= \frac{1}{2}S^{-T}S^{-1}SL \\ &= \frac{1}{2}S^{-T}L\end{aligned}$$

□

We now introduce the reduced simplex gradient. We emphasize that in contrast to the standard simplex gradient, this only requires specifying a sample set of $d+1$ points and implicitly converts this set into a centered set, simplifying later discussion.

Definition 48. *Let $\mathcal{X}$ be a sample set with $d+1$ elements satisfying Assumption 2. The reduced centered simplex gradient is given by*

$$\nabla_S^{(R)}\tilde{f}(x_0) = \frac{1}{2}S^{-T}L\tilde{\mathbf{f}}_{\mathcal{X}_L}. \tag{4.1.1}$$

Remark 6. *Using Proposition 47 it is immediate that $\nabla_{S_L}\tilde{f}(x_0) = \nabla_S^{(R)}\tilde{f}(x_0)$. Hence, the two formulations are equivalent.*

4.1.2 Homogeneous Polynomials and Taylor's Theorem

The simplex gradient is a local operation and hence suggests that instead of working with generic functions we approximate them via polynomials. The polynomials of interest are homogeneous polynomials as they are strongly connected to the functions induced by the derivatives of a

function. The basic definitions and setup for this have been presented in Section 2.2.2. In the following we state and prove technical results which will be central to the following discussion.

Technical Results for Homogeneous Polynomials

The following two results will be referred to throughout the paper and will also provide the basis for our main result.

Proposition 49. *Let p be a homogeneous polynomial in d variables of order k. Let $T \in \mathbb{R}^{d\times m}$ where $m \in \mathbb{Z}$. Then $p_T(\cdot) = p(T\cdot)$ is a homogeneous polynomial in m variables of order k.*

Proof. Notice that $(Tx)^\alpha = (\sum_{j=1}^m T_{1,j}x_j)_1^{\alpha_1} \cdots (\sum_{j=1}^m T_{d,j}x_j)_d^{\alpha_d}$. Which is a polynomial as it is the product of polynomials. Hence p_T is a polynomial. To show that it is homogeneous of order k consider $p_T(\lambda x) = p(\lambda Tx) = p(\lambda y)$ for $y = Tx$. Since p is homogeneous we have $p_T(\lambda x) = \lambda^k p_T(x)$. □

Proposition 50. *Let p be a homogeneous polynomial of order k with k being odd. Then there exists $x \neq 0$ such that $p(x) = 0$.*

Proof. Let v be a non-zero vector. If v is a root we are done. Otherwise $p(v) = c$ and $p(-v) = (-1)^k p(v) = -p(v)$ by homogeneity and assuming that k is odd. Define a continuous path $\gamma : [0, 1] \to \mathbb{R}^d$ such that $\gamma(0) = v$ and $\gamma(1) = -v$ which does not pass through the origin. Consider the continuous function $p(\gamma(t))$. By the intermediate value theorem there exists t' such that $p(\gamma(t')) = 0$ and $\gamma(t')$ is a non-zero root. □

4.2 Optimization Problem for Optimal Sample Set

In this Section we develop an objective function to assess the performance of a sample set for the centered simplex gradient. The objective function can then be minimized to find an optimal sample set. Since the function evaluations are stochastic, we use a mean squared error and use a Taylor approximation to motivated a simplified objective function. We then proceed with a

case-study to further argue the usefulness of this objective function as a surrogate for the true gradient estimation problem. In the case-study we also develop crucial theory for the discussion of the gradient estimation problem. Specifically, on how to assess the potential performance gains any method based on minimizing the proposed objective function can achieve. These will be fundamental for the numerical discussions in Sections 4.6 and 4.7.

Mean Squared Error Expression

The objective is to find the sample set $\mathcal{X}$ which on average will produce the best gradient estimate. Since the simplex gradient is a random quantity, due to relying on the evaluation of the black-box function $\tilde{f}$, we consider the mean squared error. To work with this error, we show that it can be decomposed into an **approximation error** and **noise error** term.

Proposition 51 (Mean Squared Error)**.** *Let the sample set $\mathcal{X}$ satisfy Assumption 2 and the black-box function $\tilde{f}$ be given with Assumption 1 on the noise. The mean squared error of the centered simplex gradient can be decomposed as follows*

$$\begin{aligned}&\mathbb{E}[\|\nabla_S^{(R)}\tilde{f}(x') - \nabla f(x')\|_2^2] \\ &= \underbrace{\|\nabla_S^{(R)} f(x') - \nabla f(x')\|_2^2}_{\textit{Approximation Error}} + \underbrace{\frac{\sigma^2}{4}\|S^{-1}\|_F^2}_{\textit{Noise Error}}\end{aligned}$$

where we take the expectation with respect to the noise ϵ.

The noise error is simple to interpret, as it depends solely on the form of the sample set via the difference set S. In a real world setting, we can therefore estimate the noise error purely by considering the form of S. However, the approximation error relies on understanding the true function f. Even if we knew the true function, developing a methodology to construct the optimal sample set which can account for all possible functional forms is difficult. Instead, we aim to approximate the true function and estimate the true approximation error.

4.2.1 Approximation Error Estimate via Polynomials

We choose to approximate the approximation error via a truncated Taylor Expansion. The choice is based on the fact that locally the first few terms of the Taylor Expansion are able to explain most of the variation in a function. The specific statement is given via Taylor's Theorem as stated in Theorem 2. Other methods, such as kernel methods, have been considered, however the choice of the kernel will matter greatly. It is then not immediately obvious how many terms are needed to sufficiently represent the true function.

Of course, in any application it is unlikely that we will have the exact third-order derivative available, as we assume here. However, to study the problem, we need to assume properties about the problem which may not be practical. In Section 4.7 we then show that these insights can be used for problems where we do not have access to the exact third-order derivatives.

Now, by using the polynomial basis and a centered simplex gradient, we obtain the following relation on the approximation error.

Proposition 52. *Let f be a five times continuously differentiable function on a compact domain D. Let $\mathcal{X}$ be a sample set satisfying Assumption 2 fully contained in D. Then the approximation error satisfies*

$$\|\nabla_S^{(R)} f(x_0) - \nabla f(x_0)\|_2^2 = \frac{1}{36}\|S^{-T} D^{\otimes 3} f|_{x_0}[\mathcal{S}]\|^2 + \mathcal{O}(R(\mathcal{S})^6) + \mathcal{O}(R(\mathcal{S})^8).$$

where $\mathcal{O}(R(\mathcal{S})^6) = \mathcal{O}(R(\mathcal{S})^4 \|S^{-1} D^{\otimes 3} f|_{x_0}[\mathcal{S}]\|)$.

Notably the $\mathcal{O}(R(\mathcal{S})^6)$ order term is controlled by $\|S^{-T} D^{\otimes 3} f|_{x_0}[\mathcal{S}]\|$. Hence, when we can make $\|S^{-T} D^{\otimes 3} f|_{x_0}[\mathcal{S}]\|$ close to zero the dominating order term will be $\mathcal{O}(R(\mathcal{S})^8)$.

We use the following Lemma to understand how each term in the Taylor expansion is affected.

Lemma 53. *Let $\mathcal{X}$ be a sample set satisfying Assumption 2. Then for any homogeneous*

polynomial p of order k we have

$$\frac{1}{2}S^{-T}Lp(\mathcal{S}_L) = \begin{cases} S^{-T}p(\mathcal{S}) & \text{if } k \text{ is odd,} \\ 0 & \text{otherwise.} \end{cases}$$

Proof. By definition of a function vector and properties of homogeneous polynomials we have $(p(\mathcal{S}_L))_i = (-1)^k(p(\mathcal{S}_L))_{2i}$ for $1 \leq i \leq d$. Applying L the result follows. □

Proposition 52. Applying Taylor's Theorem up to order five we have

$$\nabla_S^{(R)} f(x_0) = \sum_{k=0}^{4} \frac{1}{k!}\frac{1}{2} S^{-T} L D^{\otimes k} f|_{x_0}[\mathcal{S}_L] + \frac{1}{2} S^{-T} L R_{x,5}(\mathcal{S}_L)$$

By Lemma 53 we know that all the even terms vanish yielding

$$\begin{aligned} \|\nabla_S^{(R)} f(x_0) - \nabla f(x_0)\|^2 &= \|\frac{1}{6} S^{-T} D^{\otimes 3} f|_{x_0}[\mathcal{S}] + \frac{1}{2} S^{-T} L R_{x_0,5}(\mathcal{S}_L)\|^2 \\ &\leq \frac{1}{36}\|S^{-1} D^{\otimes 3} f|_{x_0}[\mathcal{S}]\|^2 + \frac{1}{6}\|S^{-T} D^{\otimes 3} f|_{x_0}[\mathcal{S}]\|\|S^{-1} L R_{x_0,5}(\mathcal{S}_L)\| \\ &\quad + \frac{1}{4}\|S^{-T} L R_{x_0,5}(\mathcal{S}_L)\|^2. \end{aligned}$$

The remainder term is not homogeneous as given, hence we cannot immediately extract the radius for the final result. To provide a bound we use Cauchy-Schwartz on matrix norms to obtain $\|S^{-T} L R_{x_0,5}(\mathcal{S}_L)\| \leq \|S^{-T}\|\|L R_{x_0,5}(\mathcal{S}_L)\|$. For the remainder term we use the bound from Taylor's Theorem giving us $|R_{x_0,5}(s)| \leq \frac{M}{5!}(\sum_{i=1}^{d} |s_i|)^5 =: R_{x_0,5}^{(ub)}(s)$. For completeness we note that a similar argument leads to $|D^{\otimes 3} f|_{x_0}[s]| \leq r^3 \frac{N}{3!}(\sum_{i=1}^{d} |s_i|)^3$ where $N = \sup\{|\partial^\alpha f(x_0)| : |\alpha| = 3, x \in D\}$. Using this bound we obtain $\|L R_{x_0,5}(\mathcal{S}_L)\| \leq 2\|R_{x_0,5}^{(ub)}(\mathcal{S})\|$.

Finally, using the homogeneity of $D^{\otimes 3} f|_{x_0}$ and $R_{x_0,5}^{(ub)}$ and letting $S = R(\mathcal{S})\hat{S}$ we have

$$\begin{aligned} \|\nabla_S^{(R)} f(x_0) - \nabla f(x_0)\|^2 \leq \frac{R(\mathcal{S})^4}{36}\|\hat{S}^{-1} D^{\otimes 3} f|_{x_0}[\hat{\mathcal{S}}]\|^2 \\ + \frac{R(\mathcal{S})^6}{3}\|\hat{S}^{-T} D^{\otimes 3} f|_{x_0}[\hat{\mathcal{S}}]\|\|\hat{S}^{-1}\|_2\|R_{x_0,5}^{(ub)}(\hat{\mathcal{S}})\| \\ + R(\mathcal{S})^8\|\hat{S}^{-T}\|_2^2\|R_{x_0,5}^{(ub)}(\hat{\mathcal{S}})\|^2. \end{aligned}$$

□

4.2.2 Optimization Problem

We are trying to solve the following optimization problem, which is based on the motivation above. For the formulation, we use homogeneous polynomials, with the immediate understanding that there is a one-to-one relationship to the forms defined by derivatives of a function as discussed in Section 4.1.2. Using homogeneous polynomials disentangles the problem formulation from the idea that we are required to use exact higher-order derivative information. We define the problem on the extended real-number line as it is then well-defined as a minimization problem.

Definition 54 (Objective Function)**.** *Let p be a homogeneous polynomial of degree three, $\sigma \in \mathbb{R}_+$ and $h \in \mathbb{R}_+$. Let*

$$\boldsymbol{AE} = \frac{1}{36}\|S^{-T}[p(s_1), \ldots, p(s_d)]^T\|_2^2$$

be the approximation error and

$$\boldsymbol{NE} = \frac{1}{4}\sigma^2\|S^{-1}\|_F^2$$

be the noise error. We then define the function $l_{p,\sigma,h} : \mathbb{R}^{d\times d} \to [0, \infty]$ as

$$\ell_{p,\sigma,h}(S) = \begin{cases} \boldsymbol{AE} + \boldsymbol{NE} & \text{if } h \geq R(\mathcal{S}) \text{ and } S \in GL_d \\ \infty & \text{, else.} \end{cases}$$

4.2.3 Solution when the difference matrix is orthogonal, Approximation and Noise error trade-off

We consider the case of finding the optimal orthogonal sample set. More concretely, we write the difference matrix as $S_\Lambda = U\Lambda$, where U is a fixed orthogonal matrix and Λ is a diagonal matrix with $\Lambda_i = \sqrt{\lambda_i}$ and $\lambda_i \geq 0$ for $1 \leq i \leq d$. This sheds important insight into the trade-off between

noise and approximation error. The famous example of such a centered simplex gradient is central differences.

Proposition 55. *Let p be a degree three homogeneous polynomial, U be a given orthogonal matrix and $h, \sigma \in \mathbb{R}_+$. Letting $M_i = p(u_i)$ and $S_\Lambda = U\Lambda$. We then have that Λ^*, such that $S(\Lambda^*) = \arg\min_{\Lambda^* \in \mathbb{R}^{d\times d}} \ell_{p,\sigma,h}(S_\Lambda)$, is given by*

$$\Lambda_i^* = \min\left(h, \frac{3\sigma}{|M_i|\sqrt{2}}^{1/3}\right)$$

with the loss given by

$$\begin{aligned}\ell_{p,\sigma,h}(S(\Lambda^*)) &= \sum_i \frac{1}{36}(\lambda_i^*)^2 M_i^2 + \frac{\sigma^2}{4}\frac{1}{\lambda_i^*} \\ &\geq \sum_{i=1}^{d}\left(\frac{\sqrt{3}}{16}|M_i|\sigma^2\right)^{2/3},\end{aligned}$$

with the lower-bound achieved when $\Lambda_i^* = \frac{3\sigma}{|M_i|\sqrt{2}}^{1/3}$ for $1 \leq i \leq d$.

Proof. To have finite objective function value, we enforce that no element on the diagonal of Λ is zero and all elements are less than h in absolute value. Given this and assuming that $S = U\Lambda$, we first expand the approximation error and then the noise error. For the approximation error, by the homogeneity of p we have $p(s_i) = \Lambda_i^3 M_i$ for $1 \leq i \leq d$. Furthermore, we have $S^{-1}(S^{-1})^T = (\Lambda^{-1})^2$. Hence, we have $\|(S^{-1})^T p(\mathcal{S})\|^2 = \sum_{i=1}^d \lambda_i^2 M_i^2$. For the noise error, we have that $\|S^{-1}\|_F^2 = \sum_{i=1}^d \frac{1}{\lambda_i}$ by the definition of the Frobenius norm.

In combination, we see that the form of the objective function is $\ell_{p,\sigma,h}(S) = \sum_{i=1}^d \frac{1}{36}\lambda_i^2 M_i^2 + \frac{\sigma^2}{4}\frac{1}{\lambda_i}$. Since the Λ terms appear as powers of two, it suffices to consider $\Lambda \succ 0$. We find by setting $\partial_{\Lambda_i}\ell_{p,\sigma,h}(S_\Lambda) = 0$ that the extremum of Λ_i is at $\frac{3\sigma}{|M_i|\sqrt{2}}^{1/3}$. Since the function is convex, the optimal value is obtained by taking the minimum of the extremum and h. Plugging the minimizer into $\ell_{p,\sigma,h}(S_\Lambda)$ completes the proof. □

The above provides some valuable intuition for the gradient estimation problem. Each optimal difference vector length is achieved through a balance of the approximation error and the noise

error, that is, the ratio $\frac{\sigma}{M}$. The larger the noise error relative to the approximation error, the larger the length has to be. However, it is clear via the lower-bound on the loss that increasing either quantity will decrease the performance of the gradient estimator as the problem is made more difficult. We now consider a case-study where higher-order errors are included. Through the study we can better understand how the objective function given above relates to the gradient estimation problem.

4.2.4 Case Study: Gradient Estimation

We want to study a more complete description of the gradient estimation problem and see how well the objective function given in Definition 54 approximates it. To make this more concrete, consider a fifth order polynomial $p(x) = \sum_{k=1}^{5} p^{(k)}(x)$. Notice that $\frac{1}{k!} D^{\otimes k} p|_{\mathbf{0}}[s] = p^{(k)}(s)$. While we provide a specific function p, notice that the above can also be viewed as an upper-bound on the error for a generic function f as given in Proposition 52.

As we study the problem, we require a notion of the average size of each homogeneous polynomial. We let $M^{(k)} \approx p^{(k)}(x)$ for $\|x\| = 1$. One interpretation of this approximation is as an upper-bound on the polynomial in the unit ball. The other interpretation is that $M^{(k)}$ is the value of an average polynomial at an average unit distance. The latter interpretation is made precise in Section 4.6 where we construct such polynomials. An assumption we make often throughout the case study is that we only have access to the average size of the fifth-order polynomial rather than the full structure. This imitates the set-up of the objective function where, besides constraining the size of the sample set, we optimize agnostic to higher-order structures. We will consider the impact of this assumption throughout the case study.

Letting $S = R(\mathcal{S})\hat{S}$, using the homogeneity property, and using the results from Propositions 51 and 52 we have for a given sample set S satisfying Assumption 2 that

$$\mathbb{E}[\|\nabla_S^{(R)} \tilde{p}(\mathbf{0}) - \nabla p(\mathbf{0})\|^2] = R(\mathcal{S})^4 (M_{\hat{S}}^{(3)})^2 + 2R(\mathcal{S})^6 (M_{\hat{S}}^{(3,5)}) + R(\mathcal{S})^8 (M_{\hat{S}}^{(5)})^2 + \frac{1}{R(\mathcal{S})^2} \tau_{\hat{S}}.$$

where we define the terms $(M_{\hat{S}}^{(3)})^2 = \|\hat{S}^{-1} p^{(3)}(\hat{\mathcal{S}})\|^2$, $M_{\hat{S}}^{(3,5)} = 2\langle \hat{S}^{-1} p^{(3)}(\hat{\mathcal{S}}), \hat{S}^{-1} p^{(5)}(\hat{\mathcal{S}}) \rangle$, $(M_{\hat{S}}^{(5)})^2 =$

$\|\hat{S}^{-1}p^{(5)}(\hat{\mathcal{S}})\|_2^2$, and $\tau_{\hat{S}} = \frac{\sigma^2}{4}\|\hat{S}^{-1}\|_F^2$.

We can now think of this as an optimization problem with respect to the radius and structure

$$\min_{\hat{S},R(\hat{\mathcal{S}})=1} \min_{r>0} r^4(M_{\hat{S}}^{(3)})^2 + 2r^6 M_{\hat{S}}^{(3,5)} + r^8(M_{\hat{S}}^{(5)})^2 + \frac{1}{r^2}\tau_{\hat{S}} := \ell_{p,\sigma}^{(GE)}(r,\hat{S}). \tag{4.2.1}$$

We then let $r_{\hat{S}}^* := \min_r \ell_{p,\sigma}^{(GE)}(r,\hat{S})$, i.e. the optimal radius for a given structure $\hat{S}$. An immediate observation is that when $p^{(5)}(x) = 0$, then this problem reduces to the objective function given in Definition 54 when we set h to infinity.

The problem is generally not easy to interpret. It can be difficult to understand how the structure $\hat{S}$ affects the error. In the following case study, we consider various properties of this problem.

A simplifying assumption we make is that $M_{\hat{S}}^{(3,5)} = \zeta_{\hat{S}} M_{\hat{S}}^{(3)} M_{\hat{S}}^{(5)}$ where $1 \geq \zeta_{\hat{S}} \geq 0$. Due to Cauchy-Schwarz, we have $M_{\hat{S}}^{(3)} M_{\hat{S}}^{(5)} \geq M_{\hat{S}}^{(3,5)} \geq -M_{\hat{S}}^{(3)} M_{\hat{S}}^{(5)}$. Only very negative values of $M_{\hat{S}}^{(3,5)}$ affect the following analysis. These values are unlikely since $M_{\hat{S}}^{(3,5)} \approx 0$ often. This happens when we sample two uncorrelated odd-degree polynomials and take an inner-product. This argument will be made rigorous in Section 4.6. Furthermore, we will often assume that we do not have the exact structure of $p^{(5)}$ and hence cannot actively use the cross-term to reduce the gradient-estimation error.

Gradient Estimation as a Multi-Objective Problem: Assessing Dominating Errors

We first study the objective function when $\hat{S}$ is fixed. For this we introduce two central quantities for the gradient estimation problem.

Definition 56. *Let p be given and $\hat{S}$ satisfy Assumption 2. We then let $\tilde{h}_{\hat{S}} = (\frac{\tau_{\hat{S}}}{2(M_{\hat{S}}^{(3)})^2})^{1/6}$ which is the optimal radius when $p^{(5)}$ is set to zero. Similarly, we let $h_{\hat{S}} = (\frac{\tau_{\hat{S}}}{4(M_{\hat{S}}^{(5)})^2})^{1/10}$ which is the optimal radius when $p_{\hat{S}}^{(3)}$ is zero. We then let $h_{\hat{S}}^* = \min(\tilde{h}_{\hat{S}}, h_{\hat{S}})$.*

Using the above definition we want to fully cast the problem as a multi-objective problem. Specifically, if $\tilde{h}_{\hat{S}} = h_{\hat{S}}^*$ then the lower-order error in $M_{\hat{S}}^{(3)}$ will be dominating the objective

function and similarly for $h_{\hat{S}}$. Hence, focusing on the dominating term will accomplish most of the improvement in performance.

We first argue that in fact $h^*_{\hat{S}} \approx r^*_{\hat{S}}$. This is a summary of Lemma 57 which we defer to Section 4.8 4.8.1.

Lemma 57. *Let $\ell^{(GE)}_{p,\sigma}$ and $\hat{S}$ be given. If $h_{\hat{S}} \geq \tilde{h}_{\hat{S}}$ and letting $\gamma\tilde{h}_{\hat{S}} = h_{\hat{S}}$ we have that $\frac{|\tilde{h}_{\hat{S}} - r^*_{\hat{S}}|}{\tilde{h}_{\hat{S}}} = \mathcal{O}(\zeta_{\hat{S}}\gamma^5) + \mathcal{O}(\gamma^{10})$. When $h_{\hat{S}} \leq \tilde{h}_{\hat{S}}$ and $\gamma h_{\hat{S}} = \tilde{h}_{\hat{S}}$ we have that $\frac{|h_{\hat{S}} - r^*_{\hat{S}}|}{h_{\hat{S}}} = \mathcal{O}(\zeta_{\hat{S}}\gamma^3) + \mathcal{O}(\gamma^6)$. An absolute bound is given by $\frac{|h^*_{\hat{S}} - r^*_{\hat{S}}|}{h^*_{\hat{S}}} \leq \frac{1}{5}$.*

While the above tells us that $h^*_{\hat{S}}$ is close to $r^*_{\hat{S}}$ it remains to show that this also leads to a negligible change in the observed loss. For this, we present the following lemma.

Lemma 58. *Let $\ell^{(GE)}_{p,\sigma}$ and $\hat{S}$ be given. Then we have that*

$$\left|\frac{\ell^{(GE)}_{p,\sigma}(\tilde{h}_{\hat{S}}, \hat{S})}{\ell^{(GE)}_{p,\sigma}(r^*_{\hat{S}}, \hat{S})} - 1\right| = \mathcal{O}\left(\left(\frac{|h^*_{\hat{S}} - r^*_{\hat{S}}|}{h^*_{\hat{S}}}\right)^2\right).$$

In aggregate, we have established that considering the gradient estimation problem in two parts is a sensible approximation. More precisely, $h^*_{\hat{S}}$ tells us to a high accuracy the true optimal radius for a fixed $\hat{S}$. The further apart $\tilde{h}_{\hat{S}}$ and $h_{\hat{S}}$ are the better the approximation. This is natural, as this means that one error will more strongly dominate the problem.

We now show how modifying $\hat{S}$ will affect the problem via the following Lemma.

Lemma 59. *Given $\ell^{(GE)}_{p,\sigma}$ and $\hat{S}$ which satisfies Assumption 2 we have that*

$$\ell^{(GE)}_{p,\sigma}(h^*_{\hat{S}}, \hat{S}) = \mathcal{O}\left(\frac{\tau_{\hat{S}}}{(h^*_{\hat{S}})^2}\right).$$

Given this result, the dependence on $\hat{S}$ is via the noise error and $h^*_{\hat{S}}$. While the noise error occurs explicitly, the approximation error only occurs via $h^*_{\hat{S}}$. Interestingly, when modifying $\hat{S}$ only the smaller of $\tilde{h}_{\hat{S}}$ and $h_{\hat{S}}$ will have an affect on the loss. Hence, given a $\hat{S}$ we focus solely on reduce the dominating error. We follow now with a result on the best case improvements.

Best Case Improvement Compared to Baseline

We study the best case improvement to a baseline. The set-up will resemble the assumptions we also make for the objective function given in Definition 54. For this we assume that by varying $\hat{S}$ we are able to modify $p^{(3)}$ but assume that $p^{(5)}$ remains constant. We will then see that this is also a good approximation for most cases.

The baseline sample set is an orthogonal sample set, which we denote by $\underline{\hat{S}}$. We also assume that for all sample points x in the baseline $p^{(3)}(x) = M^{(3)}$ and $p^{(5)}(x) = M^{(5)}$. This indicates an upper-bound or an average size of the polynomials, since for a baseline we assume to have no knowledge of the structure of the polynomial yet. Since $\underline{\hat{S}}$ is orthogonal we have $M^{(3)}_{\underline{\hat{S}}} = \sqrt{d}M^{(3)}$, $M^{(5)}_{\underline{\hat{S}}} = \sqrt{d}M^{(5)}$ and $\tau_{\underline{\hat{S}}} = \frac{d\sigma^2}{4}$. This sample set also minimizes the noise error among all sample sets with radius one as we will see in Lemma 64.

For this case the optimal radii simplify significantly and will be useful for intuition throughout the text. Specifically, we have $\tilde{h} = \left(\frac{\sigma^2}{8(M^{(3)})^2}\right)^{1/6}$ and $h = \left(\frac{\sigma^2}{16(M^{(5)})^2}\right)^{1/10}$. Notice we dropped the subscript because h and $\tilde{h}$ is only in terms of central quantities of the given gradient estimation problem. We will shortly see that the given h here and the hyperparameter in Definition 54 are in a certain sense equivalent.

The best case improvement is then given by a sample set $\hat{S}^*$ which minimizes the noise error and fully removes the error due to $M^{(3)}_{\hat{S}}$. We cannot remove the higher-order error in $M^{(5)}_{\hat{S}}$ since we assume to be agnostic to it. Due to the later Lemma 64 $\hat{S}^*$ is assumed to be orthogonal. Due to these assumption it remains that $h_{\hat{S}^*} = h$.

Lemma 60. *Let $\underline{\hat{S}}$ be a baseline sample set and $\hat{S}^*$ be the optimal sample set as described above. Assume that $\tilde{h} \leq h$ and $\tilde{h} = \gamma h$ then*

$$\frac{\ell^{(GE)}_{p,\sigma}(\tilde{h}, \underline{\hat{S}})}{\ell^{(GE)}_{p,\sigma}(h, \hat{S}^*)} = \mathcal{O}\left(\frac{1}{\gamma^2}\right).$$

Assume $\tilde{h} \geq h$ *and* $h_{\hat{S}} = \gamma \tilde{h}_{\hat{S}}$ *then we have*

$$\left| \frac{\ell_{p,\sigma}^{(GE)}(h, \underline{\hat{S}})}{\ell_{p,\sigma}^{(GE)}(h, \hat{S}^*)} - 1 \right| = \mathcal{O}(\gamma^6) + \mathcal{O}(\zeta_{\underline{\hat{S}}} \gamma^3).$$

Interestingly, to understand the best-case improvement we solely need the ratio of h and $\tilde{h}$. Hence, these quantities can be used to describe the problem properties as we will also do in the Numerics section. Of course, even if two problems have the same ratio of h and $\tilde{h}$, depending on the specific problem we may or may not be able to achieve this performance.

When $\tilde{h} \leq h$ the improvement that can be achieved is substantial, especially as $\tilde{h} \leq h$. We will see that in many cases we can achieve this performance or get close to it. Even if we cannot fully achieve this, even getting close to it can significantly improve performance. On the other hand, when $\tilde{h} \geq h$ it is not worth-while to seek $\hat{S}^*$ as the baseline will perform just as good.

Validity of the Objective Function given in Definition 54

The goal of the optimization problem is to get as close as possible to the best-case improvement under the same assumptions. Specifically, we assume to optimize while being agnostic to how $p^{(5)}$ varies with the sample set, that is letting $p^{(5)}(\hat{S}) = M^{(5)}\mathbf{1}$. We argue that optimizing with a constant $p^{(5)}$ is similar to finding the minimum of the objective function in Definition 54 while letting h be defined as in the previous section. Of course, if h is simply tuned as a hyperparameter, there exists a value for which the optimization problems overlap exactly. Hence, this section simply focuses on showing that h can have the interpretation of the optimal radius as given above.

Given this assumption and expanding, we have that $h_{\hat{S}} = h \left(\frac{\|\hat{S}^{-1}\|_F^2}{\|\hat{S}^{-T}\mathbf{1}\|^2} \right)^{1/10}$. The first observation is that for any $\hat{S}$ that is orthogonal, potentially not orthonormal, $\|\hat{S}^{-1}\|_F^2 = \|\hat{S}^{-T}\mathbf{1}\|^2$ as both reduce to the sum of the eigenvalues. Hence, for methods which deal with orthogonal sets, such as the one introduced in Proposition 55, the upper-bound will be appropriate.

We now provide an error rate for how the gradient estimation error behaves when $h \neq h_{\hat{S}}$.

We recall that $\frac{d}{\lambda_{max}} \leq \|\hat{S}^{-T}\mathbf{1}\|^2, \|S^{-1}\|_F^2 \leq \frac{d}{\lambda_{min}}$. Letting the condition number be given by $\kappa = \frac{\lambda_{max}}{\lambda_{min}}$ we have $h\frac{1}{\kappa^{1/10}} \leq h_{\hat{S}} \leq h\kappa^{1/10}$. To understand the effect of using h instead of $h_{\hat{S}}$ on the loss consider $\frac{\ell_{p,\sigma}^{(GE)}(\min(h,\tilde{h}_{\hat{S}}),\hat{S})}{\ell_{p,\sigma}^{(GE)}(r_{\hat{S}}^*,\hat{S})}$, which we can rewrite as $\frac{\ell_{p,\sigma}^{(GE)}(\min(h,\tilde{h}_{\hat{S}}),\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}^*,\hat{S})}\frac{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}^*,\hat{S})}{\ell_{p,\sigma}^{(GE)}(r_{\hat{S}}^*,\hat{S})}$. Recall, from previous Sections that we already have a bound on $\frac{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}^*,\hat{S})}{\ell_{p,\sigma}^{(GE)}(r_{\hat{S}}^*,\hat{S})}$. So, we now provide a Lemma for the growth of $\frac{\ell_{p,\sigma}^{(GE)}(\min(h,\tilde{h}_{\hat{S}}),\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}^*,\hat{S})}$.

Lemma 61. *When $h \geq \tilde{h}_{\hat{S}}$ we have that $\frac{\ell_{p,\sigma}^{(GE)}(\min(h,\tilde{h}_{\hat{S}}),\hat{S})}{\ell_{p,\sigma}^{(GE)}(r_{\hat{S}}^*,\hat{S})} = 1$. When $h \leq \tilde{h}_{\hat{S}} \leq h_{\hat{S}}$ we have $\frac{\ell_{p,\sigma}^{(GE)}(h,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}^*,\hat{S})} = \mathcal{O}(\gamma^2\kappa^{1/5})$. When $h \leq h_{\hat{S}} \leq \tilde{h}_{\hat{S}}$ we have $\frac{\ell_{p,\sigma}^{(GE)}(h,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}^*,\hat{S})} = \mathcal{O}(\kappa^{1/5})$. Lastly, when $h_{\hat{S}} \leq h \leq \tilde{h}_{\hat{S}}$ we have $\frac{\ell_{p,\sigma}^{(GE)}(h,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}^*,\hat{S})} = \mathcal{O}(\kappa^{4/5})$.*

Note, when $\tilde{h}_{\hat{S}} \leq h$ there is no change in the error. So, for all $\hat{S}$ for which this is true the proposed objective function is appropriate. While when $\tilde{h}_{\hat{S}} \geq h$ there may be some error associated, we have shown in the above Lemma that the error grows mostly slowly with an increased condition number. Furthermore, in these cases the problem exhibits a bias to sample sets with low condition number. Consider, the cases where $h_{\hat{S}} \leq \tilde{h}_{\hat{S}}$, which also captures the case where the error grows almost linearly. By Lemma 59, modifying $\hat{S}$ we see that there is no gain from further increasing $\tilde{h}_{\hat{S}}$ and hence the noise error will dominate. Since the noise error is minimized when the sample set is orthogonal this will bias the optimal sample set $\tilde{S}^*$ to have a lower condition number. The case when $h \leq \tilde{h}_{\hat{S}} \leq h_{\hat{S}}$ reduces to the above when $\gamma \approx 1$. When $\gamma \ll 1$ the error is also minimal except in cases with extreme condition numbers. These cases will only be achieved if $\tilde{h} \ll h$ as otherwise a orthogonal sample-set would suffice as seen in the previous section. Hence, we may conclude that in many cases the error incurred from using a fixed upper-bound of h is insignificant compared to the improvement that can be gained.

What is left is to argue that it is worth-while to study the simplified objective function in Definition 54, that is $\tilde{h} \ll h$. To argue this we claim that for many problems it is unlikely that $\tilde{h} \approx h$. For this note that $\tilde{h} = h$ when

$$M^{(5)} = \sqrt{2}\left(\frac{(M^{(3)})^5}{\sigma^2}\right)^{1/3}.$$

There are two observations. For one, for a given noise level $M^{(5)}$ will grow as roughly the square of $M^{(3)}$. Hence, the size of $p^{(5)}$ has to increase significantly more. The other insight is that $M^{(5)}$ is inversely proportional to the noise error. This makes intuitive sense, as for example with low-noise errors the optimal radii will be smaller. In regions of smaller radius the lower-order terms dominate. On the other hand, as the noise-error increases, so will the radii, which is where higher-order errors dominate.

Overall, the choice of the approximation of the gradient estimation problem as given in Definition 54 will be problem dependent. Nevertheless, it appears reasonable given the above calculations for a large set of problems the assumption that $\tilde{h} \ll h$ is reasonable and hence the approximation used for this paper is appropriate.

4.3 Global minimum: d orthogonal roots do not always exist

In this Section we study the global minimum problem. We first show that the structure of the global minimum is given by a set of d orthogonal roots. We then briefly discuss its relation to other fields in mathematics, specifically algebraic geometry, and the immediate challenges faced with solving such a problem. We then prove that such a set of d orthogonal roots cannot always exist for every homogeneous polynomial. While we can not guarantee to find the global minimum, we will in the next Section develop a methodology which uses the insights built here.

4.3.1 The Global Minimum: Orthogonal Roots

The intuition behind the gradient estimation problem is that sample points should be along directions in which the approximation error does not increase too much. These are precisely the roots of the homogeneous polynomials. Then we can increase the difference vector length and hence mitigate the noise error. The complete set of elements then should also follow an

orthogonal structure to completely minimize the noise error. We make this precise with the following proposition.

Proposition 62. *Let $S \in \mathbb{R}^{d\times d}$ be such that the following conditions are satisfied*

- $p(s_i) = 0$ *for* $1 \leq i \leq d$,
- $\langle s_i, s_j \rangle = h^2\delta_{ij}$ *for* $1 \leq i, j \leq d$,

for a homogeneous polynomial p. Then S is a global minimum of the objective function 54. Furthermore, the lower bound on the objective function is given by

$$\ell_{p,\sigma,h} = \frac{1}{4}\sigma^2 d\frac{1}{h^2}.$$

This lower bound is achieved if and only if S satisfies the above conditions.

We prove this statement by introducing two lemmas. One will tell us that the approximation error requires roots, the other will discuss that orthogonality is important for the noise error.

Lemma 63. *The approximation error is zero if and only if s_i are roots of the homogeneous polynomial p.*

Proof. For the following, assume that S is invertible and $R(S) \leq h$ so that the objective function value is finite.

For the approximation error, notice that $\|(S^{-1})^T p(\mathcal{S})\|^2 \geq 0$. A sufficient condition for this lower bound to be met is $p(s_i) = 0$ for $1 \leq i \leq d$. However, this is also necessary, as S is invertible by assumption and hence has empty kernel. □

The next is more elaborate and tells us that orthogonality is important to minimize noise error.

Lemma 64. *For S satisfying Assumption 2 the noise error, given by $\|S^{-1}\|_F^2$, under the constraint $R(\mathcal{S}) \leq h$ is minimized if and only if $S^T S = h^2 I_{d\times d}$.*

Proof. For the objective function to be finite, we assume that S is invertible and $R(S) \leq h$. By properties of the Frobenius norm, we have that $\|S\|_F^2 = \sum_{i=1}^d \lambda_i$ where the set $\{\lambda_i\}_{1\leq i\leq d}$ is the set of real non-negative eigenvalues of S^TS. Furthermore, $\|S\|_F^2 = \sum_{i=1}^d \|s_i\|^2$ which has to be less than dh^2 by assumption that $R(S) \leq h$. Hence, we have the constraint on the eigenvalues that $\frac{1}{d}\sum_{i=1}^d \lambda_i \leq h^2$.

We now solve the problem of finding S such that $\sum_{i=1}^d \frac{1}{\lambda_i}$ under the constraint $\frac{1}{d}\sum_{i=1}^d \lambda_i \leq h^2$ is minimized. We used the fact that $\|S^{-1}\|_F^2 = \sum_{i=1}^d \frac{1}{\lambda_i}$. Notice that the feasible region contains the region constrained by $R(S) \leq h$. Hence, any solution to this problem which also satisfies $R(S) \leq h$ is also a minimizer for the original problem. Furthermore, they are then also the only solutions to the original problem as $R(S) \leq h$ is strictly contained in the set of all S such that $\frac{1}{d}\sum_{i=1}^d \lambda_i \leq h^2$.

Writing the Lagrangian we have $\mathcal{L}(\lambda_1, \ldots, \lambda_d, \mu) = \sum_{i=1}^d \frac{1}{\lambda_i} + \mu(\sum_{i=1}^d \lambda_i - h^2)$. Taking the partial derivative with respect to λ_i and setting to zero we have that $\frac{1}{\lambda_i^2} = \frac{\mu}{d}$ for $1 \leq i \leq d$. This implies that $\lambda_i = \lambda_j$ for $1 \leq i, j \leq d$. Using the constraint we have immediately that $\lambda_i = h^2$.

Since every eigenvalue of S^TS is the same, $S^TS = h^2 I_{d\times d}$. Any S satisfying this condition is clearly a valid solution to minimizing the noise error under $R(S) \leq h$. □

Proposition 62. Combine Lemmas 63 and 64. □

4.3.2 Context and First Challenges

While for even degree homogeneous polynomials the prospect of finding a set of d orthogonal roots is unlikely to be fruitful, for odd degree polynomials there is more hope. For even degree homogeneous polynomials, there are obvious counter-examples where there do not even exist any roots, such as for positive-definite quadratic forms. However, by Lemma 50 there exist infinitely many roots for odd degree homogeneous polynomials. Hence, it may be sensible to study the properties of such polynomials in slightly more generality, identify any useful tools and potential challenges.

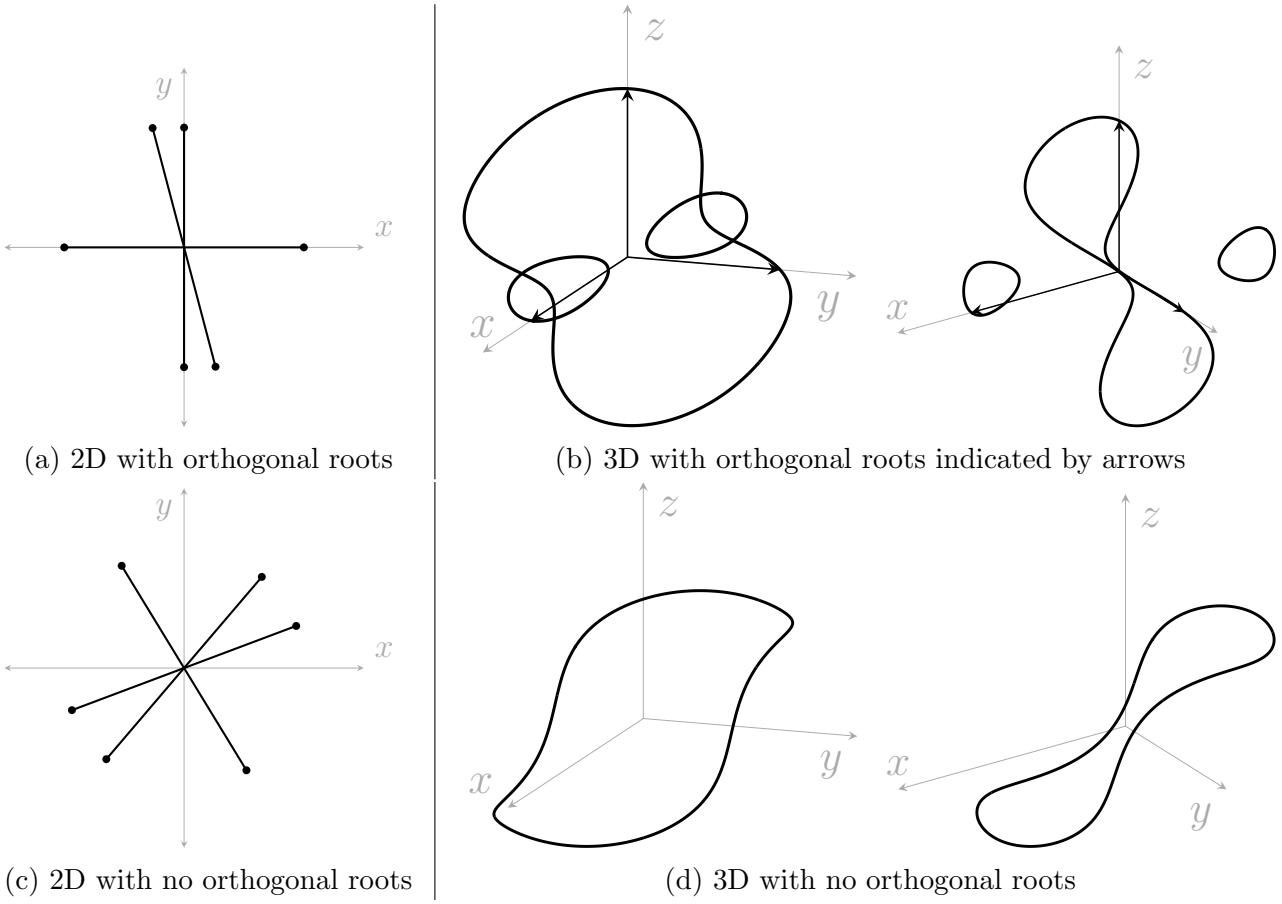
The study of roots of real polynomials is tackled by real algebraic geometry. While the question of whether a set of orthogonal roots exist for homogeneous polynomials may be of independent interest, we solely focus on solve-ability. For this, we note that the global minimum problem can be fully reformulated as finding the roots of a set of polynomial equations.

We now outline the reformulation of the global minimum problem as a solution to a system of polynomial equations over d^2 variables $\{s_i \in \mathbb{R}^d\}_{1\leq i\leq d}$. The condition $\{p(s_i) = 0\}_{1\leq i\leq d}$ ensures that all s_i are roots, $\{\sum_{l=1}^{d}(s_i)_l(x_j)_l = 0\}_{1\leq i,j\leq d, i\neq j}$ ensures orthogonality, $\{\sum_{l=1}^{d}(s_i)_l^2 - 1 = 0\}_{1\leq i\leq d}$ ensures that the zero vector is not included and makes the problem non-degenerate. The union of all polynomials then provides the system of equations that fully defines the global minimum problem. [2] The condition only prevents degenerate solutions. Simple counting shows that there are $d + d(d+1)/2$ polynomial equations that need to be satisfied.

Based on standard results, over a closed algebraic field, this problem will always have a solution for $d > 2$, as there will be more equations than variables [Har13]. However, we are working over the reals and such a solution cannot be guaranteed immediately. Consider Figure 4.2 where we show instances of roots of homogeneous polynomials. We specifically demonstrate instances where d orthogonal roots exist and some for which they do not.

However, there are results in real algebraic geometry which may allow us to tackle this problem. Specific methods, such as cylindrical algebraic decomposition, based on the Tarski-Seidenberg Theorem [SB16]. For polynomial systems over the reals, these methods can both decide whether solutions exist and find them. However, the key difficulty is that these algorithms have prohibitive doubly exponential runtime. Even for moderate problems of three and four dimensions, such a method will not work as it will require one to solve a polynomial system over a quadratic number of variables. Hence, further study of the problem is required in order to develop a methodology with a feasible runtime.

[2] The last property, making the roots have unit norm, does not constrain the problem. This is because the polynomials are homogeneous and a scaled root remains a root.



(a) 2D with orthogonal roots

(b) 3D with orthogonal roots indicated by arrows

(c) 2D with no orthogonal roots

(d) 3D with no orthogonal roots

Figure 4.2: Visualization of the set of unit roots for two and three dimensional polynomials. For the three dimensional case we show the same set of roots but from perspectives rotated by some amount.

4.3.3 There do not always exist d orthogonal roots

While at this point it may appear premature to provide a negative result, we do so as it aids later exposition. Also, the results are beneficial for studying the properties of homogeneous polynomials and provide the tools needed to study the problem in depth numerically in Section 4.6.

The main result for this section states that there do not always exist d orthogonal roots for a homogeneous polynomial of any order. In fact, for any distribution over homogeneous polynomials of a given order, there exists a new one which has a positive probability of sampling a polynomial that does not have d orthogonal roots. Unfortunately, this is mostly an existence proof and we can not methodologically construct polynomials which are guaranteed to have

this property a-priori. The lead-up to this is somewhat technical and builds all the necessary machinery.

Introduce the linear transformation

We now introduce a transform which will allow us to transform a polynomial to increase the sensitivity.

Definition 65. *Let $a \in \mathbb{R}_+$ and $v \in \mathbb{R}^d$ be a unit vector. Then we let the transform $T_{a,v}$ be given by*

$$T_{a,v}(x) = (I + (a-1)vv^T)x.$$

Furthermore, given a homogeneous polynomial $p : \mathbb{R}^d \to \mathbb{R}$ of order k we let $p_{a,v}(x) = p(T_{a,v}(x))$.

We immediately not that by Proposition 49 the newly defined polynomial $p_{a,v}$ remains a homogeneous polynomial of order k.

While the inverse of $T_{a,v}$ will be crucial, the following result states that any results on the transform also apply to its inverse.

Lemma 66. *The inverse of $T_{a,v}$ is $T_{1/a,v}$.*

Proof. We require $T_{a,v}T_{1/a,v} = I$ which is given due the following steps

$$\begin{aligned}(I + (a-1)vv^T)(I + (1/a-1)vv^T) &= I + ((a-1) + (1/a-1) + (a-1)(1/a-1))\, vv^T \\ &= I\end{aligned}$$

as $(a-1)(1/a-1) = (1-a) + (1-1/a)$. □

We now present the geometric effect of the transform.

Lemma 67. *Let $x \in \mathbb{R}^d$. The norm is given by $\|T_{a,v}(x)\| = \sqrt{\|x\|_2^2 - \langle v, x\rangle^2 + a^2\langle v, x\rangle^2}$.*

If x and v are orthogonal to each other then the same holds true for $\langle T_{a,v}(x), v\rangle$. Otherwise, the angle is derived from $\frac{\langle T_{a,v}(x),v\rangle}{\|T_{a,v}(x)\|} = (\sqrt{1+\frac{1}{a^2}(\frac{\|x\|^2}{\langle v,x\rangle^2}-1)})^{-1}$.

Proof. For the length of $T_{a,v}(x)$ we write

$$\begin{aligned}\|T_{a,v}(x)\|_2^2 &= \|x\|_2^2 + \left(2(a-1)+(a-1)^2\right)\langle v,x\rangle^2 \\ &= \|x\|_2^2 - \langle v,x\rangle^2 + a^2\langle v,x\rangle^2\end{aligned}$$

since $(a-1)^2 = a^2 - 2a + 1$.

We now compute the angle of $T_{a,v}(x)$ to v. For this we need to compute $\frac{\langle T_{a,v}(x),v\rangle}{\|T_{a,v}(x)\|}$. Notice that $\langle T_{a,v}(x), v\rangle = a\langle v,x\rangle$. If x and v were already orthogonal they remain orthogonal. For $\langle v,x\rangle \neq 0$ we obtain that

$$\frac{a\langle v,x\rangle}{\sqrt{\|x\|^2 + (a^2-1)\langle v,x\rangle^2}} = \frac{1}{\sqrt{1+\frac{1}{a^2}\left(\frac{\|x\|^2}{\langle v,x\rangle^2}-1\right)}}$$

which completes the proof. □

Sensitivity

We now discuss the effect of the transform on the sensitivity of the polynomial. Let $x \in \mathbb{R}^d$ be some unit vector and assume that on average $|p(x)| \approx M$. We will discuss the precise meaning of this and related constructions in Section 4.6.1. Then for a unit vector y we have that $p_{a,v}(y) = \|T_{a,v}(y)\|^k p(\frac{T_{a,v}(y)}{\|T_{a,v}(y)\|})$. Hence, we have that $p_{a,v}(y) \approx \|T_{a,v}(y)\|^k M$.

By Lemma 67 we know that $\|T_{a,v}(y)\| = \sqrt{1-\langle v,y\rangle^2 + a^2\langle v,y\rangle^2}$. Clearly, if $a\langle v,x\rangle \ggg 1$ we have that $\|T_{a,v}(y)\| \approx |a\langle v,y\rangle|$. Hence, the sensitivity is roughly scaled by a factor of a^k which is enhanced the closer y is to v. Notably, for all vectors that are orthogonal to v, the transform does not have any impact.

Transform and affect on roots and structure

The transform more crucially allows us to 'move' roots and hence increase a region with no roots. The primary object with which we will work is the cone that we define by $C_{\beta,v} = \{z | \frac{\langle z,v\rangle}{\|v\|\|z\|} \geq \beta\}$. We work with β directly instead of the angle $\arccos\beta$. We will often refer to β as the angle, understanding this relation.

Cones are a good representation for sets in our input space because they inhabit the following structure. We only need to look at one element in every line. We often will only look at the unit vectors. So, if no root is found, we know that none is a root.

Now define $\beta(a) = \sqrt{1 + \frac{1}{a^2}(\frac{1}{\beta^2} - 1)}^{-1}$ which leads to $C_{\beta(a),v} \cong C_{\beta,v}$. This is because $T_{a,v}$ is a bijection by Lemma 66 and for any $x \in C_{\beta,v}$ we have $T_{a,v}(x) \in C_{\beta(a),v}$ and $y \notin C_{\beta,v}$ we have $T_{a,v}(y) \notin C_{\beta,v}$ by Lemma 67. If $a > 1$ then the angle of the cone decreases and the cone is more centered around v. The opposite occurs when $a < 1$.

Now, we are interested in studying how the roots of p and $p_{a,v}$ are related. Clearly, if x^* is a root of p we then have $T_{1/a,v}(x^*)$ is a root for $p_{a,v}$ due to $T_{1/a,v}$ being the inverse of $T_{a,v}$ as given by Lemma 66. If we then consider $C_{\beta,v} \cong C_{\beta(1/a),v}$ then if $C_{\beta,v}$ has no roots then $C_{\beta(1/a),v}$ also has no roots. So, if $a > 1$ we have increased the region of no roots for $p_{a,v}$ by increasing the angle the cone $C_{\beta(1/a),v}$ makes with v.

We now use the above construction and show immediately how it allows us to guarantee the absence of d orthogonal roots.

Proposition 68. *Let p be a homogeneous polynomial such that there exist unit-vector $v \in \mathbb{R}^d$ and value $\beta \in [0, 1)$ such that there exist no roots in $C_{\beta,v}$. Let $a > \sqrt{(d-1)/(\frac{1}{\beta^2} - 1)}$ then the polynomial $p_{a,v}$ has no set of d orthogonal roots.*

We first prove the following lemma, which tells us how large a cone has to be such that no orthogonal basis can be constructed in its complement.

Lemma 69. *Let $v \in \mathbb{R}^d$ be a unit vector. Then there do not exist d orthogonal vectors $\{b_i\}_{1 \leq i \leq d}$ such that $b_i \notin C_{\frac{1}{\sqrt{d}},v}$ for $1 \leq i \leq d$.*

Proof. Let $\{b_i\}_{1\leq i\leq d}$ be an orthogonal set. Then $\|v\|^2 = \sum_{i=1}^d \frac{1}{\|b_i\|^2}\langle v, b_i\rangle^2$. Now assume that $b_i \notin C_{\frac{1}{\sqrt{d}},v}$ for $1 \leq i \leq d$. This implies that $\sum_{i=1}^d \frac{1}{\|b_i\|^2}\langle v, b_i\rangle^2 < \|v\|^2 \sum_{i=1}^d \frac{1}{d} = \|v\|^2$ which is a contradiction. □

Proposition 68. By Lemma 69 we know that if we can ensure that in the cone $C_{\frac{1}{\sqrt{d}},v}$ there are no roots, then there cannot exist a set of d orthogonal roots. By assumption there is no root in the cone $C_{\beta,v}$. By applying the T_a transform to the input of the polynomial, we know if there are no roots in the cone $C_{\beta,v}$, then there will be no roots in $C_{\beta(1/a),v}$. We now simply need for $\beta(1/a) = \frac{1}{\sqrt{d}}$. Solving for a we obtain the result. □

While there may not exist a set of d orthogonal vectors in the complement of a cone $C_{\beta,v}$, it is easy to see that there do exist $d-1$ orthogonal vectors in the complement. Whether those are roots will have to be seen. As in later discussion we consider sets of d orthogonal vectors with the first $d-1$ vectors enforced in the complement of a given cone, we now present a general result on the relation between elements in the complement of such a subspace. We make precise the obvious observation that the larger the angle of the cone the closer the elements of the complement have to be to v.

Lemma 70. *Let W be a m-dimensional subspace such that there exist $b_i^\perp \in W^\perp$ such that $b_i^\perp \notin C_{\beta,v}$ for $1 \leq i \leq d-m$. Then the closest vector $w \in W$ to v satisfies $\frac{1}{\|w\|}\langle v, w\rangle \geq \sqrt{1-(d-m)\beta^2}$ when $\beta^2 \leq \frac{1}{d-m}$.*

Proof. Let $v = w + w^\perp$ where $w \in W$ and $w^\perp \in W^\perp$. Note that w is the closest vector to v in the subspace W. Let $\|w\|\hat{w} = w$ which then yields taking the inner-product of $\hat{w}$ with $v = \hat{w}\|w\| + w^\perp$ that $\langle \hat{w}, v\rangle = \|w\|$. Expanding $w^\perp$ in terms of $\{b_i^\perp\}_{1\leq i\leq d-m}$ and taking the inner-product with v we obtain

$$\langle v, v\rangle = \langle \hat{w}, v\rangle^2 + \sum_{i=1}^{d-m} \langle b_i^\perp, v\rangle^2.$$

Since we know that $b_i^\perp \notin C_{\beta,v}$ we have that $\langle b_i^\perp, v\rangle < \beta$ yielding the result. □

For later discussion we also require a result on the intersection of a cone with a subspace. We observe that the intersection is a cone in the subspace and has an easily interpretable angle.

Lemma 71. *Let a cone $C_{\beta,v} \subseteq \mathbb{R}^d$ and a subspace $W \subseteq \mathbb{R}^d$ be given. Let w be the closest point to v in the subspace and define $\beta_{v,w} = \frac{1}{\|w\|}\langle v, w\rangle$. If $\beta_{v,w} > \beta$ then the intersection $C_{\beta,v} \cap W$ is empty. Otherwise, the intersection is given by the cone $C^{(W)}_{\tilde{\beta},w} = \{z : z \in W,\ \frac{1}{\|w\|\|z\|}\langle w, z\rangle \geq \tilde{\beta}\}$ with $\tilde{\beta} = \frac{\beta}{\beta_{v,w}}$.*

Proof. Let w be the closest point to v in the subspace given by W. If $\frac{1}{|w|}\langle w, v\rangle < \beta$ then w is not in the cone. Furthermore, no other element in W is. To observe this consider $\|v - w\|^2 = \|v\|^2 - 2\langle v, w\rangle + \|w\|^2$. If $z \in C_{\beta,v}$ then since all positive re-scalings leave z in the cone we assume that $\|z\| = \|w\|$. This then immediately implies that z must have been closer than w since $\langle v, z\rangle \geq \langle v, w\rangle$.

Now, consider the case when w is in the cone. We show that $C^{(W)}_{\tilde{\beta},w} = C_{v,\beta} \cap W$. For this, use the following insight. Choose an arbitrary vector $z \in W$. We want to now establish a relationship between the angles of each of these points. For this, let $\theta_{v,w} = \arccos(\frac{1}{\|w\|}\langle v, w\rangle)$, $\theta_{v,z} = \arccos(\frac{1}{\|z\|}\langle v, z\rangle)$ and $\theta_{w,z} = \arccos(\frac{1}{\|w\|\|z\|}\langle w, z\rangle)$.

Recall, since w is the closest point to v in the subspace W we have that $v - w$ is orthogonal to each vector in W. Specifically, it is orthogonal to w giving us $\|w\| = \cos(\theta_{v,w})$ and $\|w - v\| = \sin(\theta_{v,w})$. We then have since $v - w \perp z$ that

$$\begin{aligned} \|z - w\|^2 + \|v - w\|^2 &= \|z - v\|^2 \\ \Rightarrow \|z\|^2 - 2\|z\|\cos(\theta_{v,w})\cos(\theta_{w,z}) + \cos(\theta_{v,w})^2 + \sin(\theta_{v,w})^2 &= \|z\|^2 - 2\|z\|\cos(\theta_{v,z}) + 1 \\ \Rightarrow \cos(\theta_{w,z})\cos(\theta_{v,w}) &= \cos(\theta_{v,z}). \end{aligned}$$

Now, we show that $C^{(W)}_{\tilde{\beta},w} = C_{v,\beta} \cap W$ recalling that $\tilde{\beta} = \frac{\beta}{\beta_{v,w}}$. Assume $z \in C_{v,\beta} \cap W$, which implies that $\cos(\theta_{v,z}) \geq \beta$. By the above relation $\cos(\theta_{w,z}) = \frac{\cos(\theta_{v,z})}{\cos(\theta_{v,w})} \geq \frac{\beta}{\beta_{v,w}} = \tilde{\beta}$ and hence $z \in C^{(W)}_{\tilde{\beta},w}$.

Now, let $z \in C^{(W)}_{\tilde{\beta},w}$ and hence $\cos(\theta_{w,z}) \geq \tilde{\beta}$. From the above relation we have that $\frac{\cos(\theta_{v,z})}{\cos(\theta_{v,w})} =$

$\cos(\theta_{w,z}) \geq \tilde{\beta} = \frac{\beta}{\beta_{v,w}}$ and hence $\cos(\theta_{v,z}) \geq \beta$. Due to this and every element in $C^{(W)}_{\tilde{\beta},w}$ also being in the subspace W we have that $z \in C_{v,\beta} \cap W$. □

Existence Results

We conclude by proving that first for any polynomial there exists such a v and β. In conclusion, we extend this to show that we can sample from a distribution with probability q. The distribution statement is important to justify later experiments where we set v but guess β.

Proposition 72. *For any non-zero homogeneous polynomial p there exists a unit vector $v \in \mathbb{R}$ and $\beta \in [0, 1)$ such that under the polynomial for $x \in C_{\beta,v}$ we have $p(x) \neq 0$.*

Proof. Since p is not a zero polynomial, there exists a v such that $p(v) \neq 0$. Due to p being a homogeneous polynomial, we can assume that v is a unit vector. Due to continuity there exists a closed ball of radius R such that for $y \in B_v(R)$ we have $p(y) \neq 0$. Let $\beta = \min_{y \in B_v(R)} \frac{\langle v,y \rangle}{\|y\|}$ which is well defined since $0 \notin B_v(R)$ and we are taking the minimum of a continuous function on a compact domain. Let $y_\beta = \arg\min_{y \in B_v(R)} \frac{\langle v,y \rangle}{\|y\|}$. We now want to show that the cone $C_{\beta,v}$ contains no roots. Due to the homogeneity of the polynomial, we only need to show that for $w \in C_{\beta,v}$ there is a rescaling which is in the ball $B_v(R)$. Note that all $z \in \mathbb{R}^d$ such that $\|z\| = \|y\|$ and $\frac{\langle v,z \rangle}{\|z\|} \geq \beta$ are in the ball $B_v(R)$. This is because $\|z - v\|^2 \leq \|y_\beta\|^2 - 2\beta\|y_\beta\| + \|v\|^2 = \|v - y_\beta\|^2 \leq R^2$. So, any element in the cone can be re-scaled to have norm $\|y_\beta\|$ and hence is not a root. □

Combining Propositions 68 and 72 we have the following Corollary.

Corollary 73. *For any non-zero homogeneous polynomial p there exists $a \in \mathbb{R}$ and $v \in \mathbb{R}^d$ such that $p_{a,v}$ has no set of d orthogonal roots.*

Clearly by the above result, there always exist polynomials which do not have d orthogonal roots. To provide a more constructive method to generate such polynomials we provide the following result. The difficulty with the above proposition is that there is no constructive way to find both v and β. Instead, we us a probabilistic construction to avoid having to estimate v. Estimating β remains difficult and will rely on heuristics.

Proposition 74. *Let $w \in \mathbb{R}^d$ be a unit vector and P be a random homogeneous polynomial of order k and let it be distributionally invariant under rotations of the input space. Specifically, for any two vectors x, x' such that $\|x\| = \|x'\|$ we have that $P(x) \sim P(x')$. Then there exist $a \in \mathbb{R}_+$ and $q \in (0, 1]$ such that with probability q there does not exist a set of d orthogonal roots for the random polynomial $P_{a,v}$.*

Proof. By Corollary 73 for any instance p of the random polynomial P there exist β and v such that the cone $C_{\beta,v}$ have no roots. To make the dependence on P clear we write β_P and v_P. Since $\beta_P < 1$ and by properties of the cumulative distribution function there has to exist β' such that with probability $q' > 0$ we have $\mathbb{P}(\beta_P \leq \beta') = q'$. For the direction v_P, since P is distributionally invariant under input rotations, we have that v_P is uniformly distributed on the unit ball.

Now, consider $\beta = \frac{1+\beta'}{2}$. We claim that if $\langle v, w\rangle \geq \beta$ we have that for $C_{\beta',w} \cap C_{\beta,w} = C_{\beta,w}$, that is, no element in the cone $C_{\beta,w}$ is a root. Let $z \in C_{\beta,w}$. Then due to the triangle inequality $\|v - z\| \leq \|v - w\| + \|w - z\|$ we have that $\langle v, w\rangle + \langle w, z\rangle - 1 \leq \langle v, z\rangle$. Now, if $\beta \leq \langle v, w\rangle, \langle w, z\rangle$ we have that

$$2\beta - 1 \leq \langle v, w\rangle + \langle w, z\rangle - 1 \leq \langle v, z\rangle$$
$$\beta' \leq \langle v, z\rangle$$

and hence $z \in C_{\beta',v}$.

Now, as long as v falls in the cone $C_{w,\beta}$ we can ensure that no element in the cone is a root. The probability of this happening is given by the area of the set $\{z | z \in C_{\beta,w}, \|z\| = 1\}$ relative to the total area of the unit ball. We denote the probability by q_2, which will be non-zero since $\beta < 1$. Hence, the probability of there being no roots in the cone $C_{\beta,w}$ is lower bounded by $q = q'q_2$. To obtain the a which achieves this cone-size simply solve for $\beta(a) = a$ where $\beta(a)$ is defined in Section 4.3.3. □

4.4 ORCAS: Finding $d-1$ orthogonal roots

We now present our main theoretical result relating to the number of orthogonal roots that can be obtained. This will provide a satisfying answer to the more general question about finding roots for homogeneous polynomials. We then introduce a method based on this insight for the construction of a sample set. While the proof for Theorem 75 is mostly constructive, we make explicit how to numerically find roots. Then, we compare our method to the centered simplex gradient. In the case where higher-order terms have significantly less influence than the current term, our improvement scales with the dimension d.

4.4.1 Main Theorem and Proof

We now provide the existence result for the maximal guaranteed set of orthogonal roots. In the previous Section we showed that any more orthogonal roots are not guaranteed. Note that the proof is constructive and will provide the basis for the numerical method finding a sample set for the centered simplex gradient.

Theorem 75. *Let p be an odd degree homogeneous polynomial with $d > 1$ variables. Then there exists a set of $d-1$ orthogonal roots.*

Proof. We prove the above by induction. Let $d = 2$ and by Proposition 50 there exists a root since we are given a homogeneous polynomial of odd degree. Now assume that $d > 2$. By Proposition 50 there exists a root x^* for the polynomial. By Gram-Schmidt, let U be an orthogonal matrix such that $Ux^* = e_1$. Now, let $\tilde{U} \in \mathbb{U}^{d-1\times d}$ be constructed by removing the first row of U. Note that $\tilde{U}\tilde{U}^T = I_{d-1\times d-1}$. By Proposition 49 the polynomial in $d-1$ variables $p_{\tilde{U}^T}(\cdot) = p(\tilde{U}^T\cdot)$ is also a homogeneous polynomial of odd order k. By the induction hypothesis, there exists a set of $d-2$ orthogonal roots $S_{d-2} = \{\tilde{x}_i\}_{1\leq d-2}$. Now, consider the set $S_{d-1} = \{\tilde{U}^T x_i\}_{1\leq i\leq d-2} \cup \{x^*\}$. Clearly, every element is a root of p. Furthermore, for $\tilde{x}_i, \tilde{x}_j \in S_{d-2}$ such that $\tilde{x}_i \neq \tilde{x}_j$ we have $\tilde{x}_i^T\tilde{U}\tilde{U}^T\tilde{x}_j = 0$ as $\tilde{U}\tilde{U}^T = I_{d-1\times d-1}$ and S_{d-2} is an orthogonal set. Lastly, for x^* and each $\tilde{x}_i \in S_{d-2}$ we have $\tilde{x}_i^T\tilde{U}x^* = 0$ since $\tilde{U}x^* = 0$ by construction of $\tilde{U}$. □

While we showed in Section 4.3.3 that we cannot have a universal statement that a set of d orthogonal roots exists, we obtain the next best result. Furthermore, the proof is remarkably simple and suggests flexibility in how these $d-1$ roots are chosen. Specifically, we invoke Proposition 50 at every step of the induction showing that a root exists. However, Proposition 50 is proved by finding a root on a path between two points which have opposite sign under the polynomial. However, there are arbitrarily many paths and hence as many roots.

4.4.2 ORCAS: Orthogonal Roots Centered (Almost) Simplex

[3] We now introduce ORCAS, based on Theorem 6, which serves both as a tool to obtain a set of samples for the simplex gradient and can serve as a dimensionality reduction tool.

Notice that in Theorem 75 we only obtain a set of $d-1$ vectors. However, we need d vectors to find the minimum of the objective function presented in Definition 54. Hence, the last vector is free to choose. Therefore, we make the choice of making the final vector orthogonal to the set of $d-1$ orthogonal roots given by Theorem 75. This way, we are guaranteed to condition the set well, in terms of being orthogonal. However, it will also clearly expose us to situations where the final orthogonal direction has high sensitivity.

Furthermore, ORCAS can be used as a dimensionality reduction method. Specifically, we can specify a stopping dimension $d^{(stop)}$, where ORCAS produces $d-d^{(stop)}$ orthogonal roots and a change of basis U. Given U, we define p_U which is now a $d^{(stop)}$-dimensional homogeneous polynomial. We can then apply the objective function defined in Definition 54 on this problem and then use the change of basis transform U to obtain the resulting sample set in the original space. We will discuss this use case in Section 4.5.

Algorithm 6 ORCAS

procedure ORCAS($p, d, h, d^{(stop)}$)
  $U \leftarrow I_{d\times d}$ ▷ Set the initial rotation matrix to the Identity
  $\mathcal{S} \leftarrow \{\}$ ▷ Difference Vectors
  while $d > d^{(stop)}$ **do**
    if $d = 1$ **then**
      $M \leftarrow p_U(1)$
      $\tilde{h} \leftarrow \left(\frac{9\sigma^2}{2M^2}\right)^{1/6}$
      $h^* \leftarrow \min(h, \tilde{h})$
      $\mathcal{S} \leftarrow \mathcal{S} \cup \{h^* U\}$ ▷ Recall when $d = 1$ the variable U stores a vector.
      $U \leftarrow 0$
    else
      $x^* \leftarrow GetRoot(p_U)$
      $\mathcal{S} \leftarrow \mathcal{S} \cup \{hUx^*\}$
      $\tilde{U} \leftarrow \text{GramSchmidt}(x^*, e_1, \ldots, e_d)[2:]$
      $U \leftarrow U\tilde{U}^T$
    end if
    $d \leftarrow d - 1$
  end while
  return $\mathcal{S}, U$
end procedure

4.4.3 Finding Root

We base our root finding method on a combination of a bisection method and a Newton-Rhapson method. The bisection method allows us to guarantee a solution close to a root. While the Newton-Rhapson method is significantly more efficient and can be shown to always converge to a root, it may also converge to the zero root, i.e. $x = \mathbf{0}$. We found that the likelihood of converging to the zero-root is drastically reduced when the starting point is close to a root. Hence, Algorithm 7 first computes a point guaranteed close to a root via the bisection method and then uses this point as the starting point for the Newton-Rhapson method. As the Newton-Rhapson may still converge close to the zero-root, it is important to normalize the solution, as any non-zero root will remain a root under normalization by the properties of homogeneous polynomials. We then compare the solutions from both methods and return the one with the lowest function value. We now discuss the implementation details of both methods.

[3]The "Almost" is a bit tongue in cheek and refers to the fact that almost all sample points are roots. Also, this sounds much better than ORCS.

Algorithm 7 Find Root

procedure FIND ROOT($p, d, N_{bis}, N_{nr}, \epsilon_{bis}, \epsilon_{nr}, tol$)
  $x_{bis} \leftarrow$ Bisection$(p, d, N_{bis}, \epsilon_{bis})$
  $x_{nr} \leftarrow$ Newton-Rhapson$(p, d, x_{bis}, N_{nr}, \epsilon_{nr})$
  if $p(x_{nr}) < p(x_{bis})$ **then**
    return x_{nr}
  else
    return x_{bis}
  end if
end procedure

Bisection Method

Algorithm 8 Bisection

procedure BISECTION($p, d, N_{bis}, \epsilon_{bis}$)
  $\gamma(\theta) \leftarrow [\cos(\theta), \sin(\theta), 0, \ldots, 0]$
  $\theta_l, \theta_u \leftarrow 0, \pi$
  $\theta_m \leftarrow (\theta_l + \theta_u)/2$
  while $p(\theta_m) > \epsilon_{bis}$ & $N_{bis} > 0$ **do**
    if $sign(p(\gamma(\theta_m))) = sign(p(\gamma(\theta_l)))$ **then**
      $\theta_l \leftarrow \theta_m$
    else
      $\theta_u \leftarrow \theta_m$
    end if
    $\theta_m \leftarrow (\theta_l + \theta_u)/2$
    $N_{bis} \leftarrow N_{bis} - 1$
  end while
  return $\gamma(\theta_m)$
end procedure

In Algorithm 8 we present our implementation of the bisection method. We use the insight that for an odd degree homogeneous polynomial p we have $p(x) = -p(x)$. Hence, to find a nonzero root, we only need to construct a path $\gamma : [0, 1] \to \mathbb{R}^d$ such that $\gamma(0) = x$ and $\gamma(1) = -x$ and γ do not pass through $\mathbf{0}$.

We then run the bisection method on the function $p(\gamma(\cdot))$ and find a $c \in [0, 1]$ such that $\gamma(c)$ is close to a root. Since we half the space at each iteration, we require $-\log_2(\delta)$ iterations to reach a root to a $\delta < 1$ tolerance.

We note that the choice of γ is arbitrary. While we choose a parametrization on a semi-circle, other paths may be constructed as well. However, we found the choice to matter little to the

final ORCAS algorithm and found the semi-circle to be most interpretable.

Newton-Rhapson Method

Algorithm 9 Newton-Rhapson

procedure NEWTON-RHAPSON($p, d, x_0, N_{nr}, \epsilon_{nr}$)
  $x \leftarrow x_0$
  while $p(x) > \epsilon_{nr}$ & $N_{nr} > 0$ **do**
    $x \leftarrow x - \frac{\nabla p(x)}{\|\nabla p(x)\|^2} p(x)$
  end while
  if $x = \mathbf{0}$ **then**
    return x_0
  else
    return $\frac{x}{\|x\|}$
  end if
end procedure

The Newton-Rhapson which we use is given in Algorithm 9. We found that when the method is initialized close to a root, it finds a root close to machine precision within only a few iterations. The method has two defining properties that we present. One is generic to the method and is presented for general motivation and the other is specific to homogeneous polynomials.

We use a Newton-Rhapson method for problems with more variables than equations. As it is less commonly studied, we briefly state its defining property which justifies its general use.

Proposition 76. *Let f be a function and x be such that $\nabla f(x) \neq \mathbf{0}$. Then the Newton-Rhapson update step given by $-\frac{\nabla f(x)}{\|\nabla f(x)\|_2^2} f(x)$ is the solution to $\min_{\Delta x} \|\Delta x\|^2$ such that $f(x) + \nabla f(x)\Delta x = 0$.*

Proof. The Lagrangian is given by $\mathcal{L}(\Delta x) = \|\Delta x\|^2 + \lambda(f(x) + \langle \nabla f(x), \Delta x \rangle)$ and the gradient when set to zero gives $2\Delta x + \lambda \nabla f(x)$ which implies that $\Delta x \propto \nabla f(x)$. The constraint then yields that $f(x) + \lambda \|\nabla f(x)\|^2 = 0$ which implies that $\lambda = -\frac{\nabla f(x)}{\|\nabla f(x)\|^2} f(x)$. □

We now state a result relating to the convergence of the method for homogeneous polynomials and the well-definedness of the update step.

Proposition 77. *Let p be a homogeneous polynomial of degree k. Then $\nabla p(x) p(x) = 0$ if and only if $p(x) = 0$.*

Proof. The direction when $p(x) = 0$ is trivial. For the other direction we use Euler's homogeneous function theorem which states that for a homogeneous function

$$\langle x, \nabla p(x) \rangle = kp(x).$$

Then clearly if $\nabla p(x) = 0$ we have $p(x) = 0$. □

Hence, the update-step is always well defined for points which are not roots. Furthermore, if the method converges, then it has converged to a root.

The issue with the method is that it may also converge to the zero root, i.e. $x = \mathbf{0}$. This is a general problem, as simply by moving closer to zero, the function value is reduced. To address this, any non-zero solution needs to be normalized to make it comparable to any other root candidates. This will allow us later to detect whether the output simply appears to be a root due to proximity to the origin. Any zero-solution is replaced with the initial starting point.

4.4.4 Runtime

The runtime at every step of ORCAS in Algorithm 6 is dominated by the constant number of evaluations of the homogeneous polynomial as part of the `GetRoot` subroutine defined in Algorithm 7. Since for any homogeneous polynomial there are $\binom{d+2}{d}$ terms in a degree three homogeneous polynomial, each evaluation is of order $\mathcal{O}(d^3)$. The iterations happen d times. Hence, the final runtime is $\mathcal{O}(d^4)$.

4.4.5 Worst-Case Performance and Comparison to Central Differences

We want to think of ORCAS and how it compares the optimum and the most used method in CD. For CD, there is no consideration for the direction of the difference vectors. Hence, in the worst case in every direction the error behaves as M. Similarly for ORCAS, the last difference

vector is fixed given the selection of the previous ones. However, there is no control over that direction, and hence could again be in the worst case direction. The crucial result of ORCAS is that it has significantly improved worst-case properties compared to central differences.

Most of the intuition will be drawn from the Case Study in Section 4.2.4. This is because the results have to be in relation to the hyperparameter h, which represents the optimal radius of the sample set if only higher-order errors mattered. That is, the error due to the third-order term vanished. We then let $\tilde{h} = \frac{3\sigma}{\sqrt{2}M}^{1/3}$ based on Proposition 55 which represents the radius of a sample set with a third-order error of M in each direction. The benefit of ORCAS is that only along one difference vector this error will be incurred. Consequently, $h^*_{\hat{S}} = \min(h, \tilde{h})$ and represents the radius which a method will truly be using.

As both ORCAS and CD are based on orthogonal sets we present the following Corollary based on Proposition 55.

Corollary 78. *Let $M = \max_{u:\|u\|=1} p(u)$ then the worst case error for central differences is*

$$\ell^{(ub)}_{p,\sigma,h}(S_{CD}) := \frac{d}{36}M^2(\tilde{h})^4 + \frac{\sigma^2}{4}\frac{d}{(\tilde{h})^2} \geq \ell_{p,\sigma,h}(S_{CD})$$

and for ORCAS is

$$\ell^{(ub)}_{p,\sigma,h}(S_{ORCAS}) := \frac{1}{36}M^2(\tilde{h})^4 + \frac{\sigma^2}{4}\left(\frac{d-1}{h^2} + \frac{1}{(\tilde{h})^2}\right) \geq \ell_{p,\sigma,h}(S_{ORCAS}).$$

Comparison to the optimum

In the best case, the lower-bound of the error is given by $\ell^{(lb)}_{p,\sigma,h} := \frac{\sigma^2}{4}\frac{d}{h^2}$. We look at the ratio of this error and the error for both ORCAS and CD.

For ORCAS we have the following ratio, where we have replaced the occurrences of M and σ with $\tilde{h}$

$$\frac{\ell^{(lb)}_{p,\sigma,h}}{\ell^{(ub)}_{p,\sigma,h}(S_{\text{ORCAS}})} = \frac{1}{2}(h^*_{\hat{S}})^4\frac{h^2}{\tilde{h}^6}\frac{1}{d} + \frac{h^2}{(h^*_{\hat{S}})^2}\frac{1}{d} + \frac{d-1}{d} \xrightarrow[d\to\infty]{} 1.$$

Hence, in large dimensions, we are approaching the optimum. How quickly will depend on h. The bigger we could have made $\tilde{h}$, if we were to remove the third-order error, the faster we would also converge to the optimum.

CD does not have this asymptotic property as it is independent of d

$$\frac{\ell^{(lb)}_{p,\sigma,h}}{\ell^{(ub)}_{p,\sigma,h}(S_{\text{CD}})} = \frac{1}{2}(h^*_{\hat{S}})^4\frac{h^2}{\tilde{h}^6} + \frac{h^2}{(h^*_{\hat{S}})^2}.$$

This ratio also tells us that if $\tilde{h}$ can be made close to h, then the error becomes significantly more balanced. If $h^*_{\hat{S}} = h$, then as $\tilde{h} \gg h$ we have that this also approaches 1.

Compare directly to each other

Intuitively, since we removed $d-1$ terms for M, we reduced the approximation error. And, we could increase the difference vector lengths for the other ones to the maximal size considering the higher-order terms as represented by h. We now take the ratio of both errors to obtain

$$\frac{\ell^{(ub)}_{p,\sigma,h}(S_{\text{CD}})}{\ell^{(ub)}_{p,\sigma,h}(S_{\text{ORCAS}})} = \frac{d(1+2\frac{\tilde{h}^6}{(h^*_{\hat{S}})^6})}{1+2\frac{\tilde{h}^6}{(h^*_{\hat{S}})^6}(\frac{(h^*_{\hat{S}})^2}{h^2}(d-1)+1)}.$$

In the case when the lower-order approximation error dominates, that is $\tilde{h} \leq h$ and $h^*_{\hat{S}} = \tilde{h}$, the ratio is given by

$$\frac{\ell^{(ub)}_{p,\sigma,h}(S_{\text{CD}})}{\ell^{(ub)}_{p,\sigma,h}(S_{\text{ORCAS}})} = \frac{3d}{3+2\frac{\tilde{h}^2}{h^2}(d-1)} \xrightarrow[d\to\infty]{} \frac{3}{2}\frac{h^2}{\tilde{h}^2}.$$

In the case when higher-order terms dominates, that is $\tilde{h} \geq h$ and $h^*_{\hat{S}} = h$, we obtain

$$\frac{\ell^{(ub)}_{p,\sigma,h}(S_{\text{CD}})}{\ell^{(ub)}_{p,\sigma,h}(S_{\text{ORCAS}})} = \frac{1+2\frac{\tilde{h}^6}{h^6}}{\frac{1}{d}+2\frac{\tilde{h}^6}{h^6}}.$$

Notice, that for any d, when $\tilde{h} \gg h$ the ratio tends to one. Hence, in this case the sample

sets will have the same performance. Even for moderately large d the ratio behaves as $1 + \frac{1}{2}\frac{h^6}{\tilde{h}^6}$ making it more clear that with an increasing $\tilde{h}$ the errors converge to each other.

4.5 Numerical Solvers

We have seen in the previous section that ORCAS offers improved worst-case performance. However, we are also constrained by the choice of making the set orthogonal. While this is a justified choice to minimize the noise error, it is natural to consider violating the orthogonality constraint to obtain a better final sample point.

So, consider a numerical method which can solve the original problem in Definition 54. Given that $d-1$ roots always exist, we may expect that the objective function may be solvable by such a method. The key difficult is that the dimensionality of the problem scales quadratically and that the solution has no worst-case guarantees. We proceed by presenting the numerical method. We conclude with introducing a combination of the numerical solver and ORCAS, where ORCAS is used as a dimensionality reduction method.

4.5.1 BFGS as a Numerical Solver

We have found that running the classic BFGS algorithm on a slight modification of the original problem provided the best results. We specifically run the BFGS method on the function defined on $S \in GL_d$

$$l^{NS}_{p,\sigma,h}(S) = \mathbf{AE} + \mathbf{NE} + \mu \sum_{i=1}^{d} (\|s_i\|^2 - h^2)^2$$

which is simply the original objective function plus a quadratic penalty term. We set $\mu = 0.1$ and find consistent results. We use the `scipy` implementation of BFGS [VGO+20].

We have tried various other methods which included explicit constraint handling. However, we found BFGS with the simple constraint to be most reliable. Once reason for this may be that

the quadratic penalty terms helps the problem be in a more stable regime. Specifically, consider a root x as the whole set of points cx for $c \in \mathbb{R}$. Then close to 0, an optimization step may cover multiple roots, while further away it may not. Hence, enforcing the sample points to be within a specific radius band may help with step-size selection. Of course, it may then be more difficult to find the exact optimal step-size when not all sample points are located at a root. However, this does not generally seem to be an issue numerically.

Lastly, to avoid the issue of no-guarantees, we also perform ORCAS simultaneously. We then return the sample set which has the lowest objective function value and use this sample set for gradient estimation. given a solution to this problem, we perform manual clipping of the difference set to enforce the constraint $R(\mathcal{S}) \leq h$.

4.5.2 ORCAS as a Dimensionality Reduction Method

Since the numerical solver may be prohibitively slow in higher dimensions we can use ORCAS as a dimensionality reduction tool. We call the method `ORCAS-NS`. We append $d^{(stop)}$ to indicate the dimension to which the problem will be reduced. The benefit is that ORCAS reduces the problem to the given dimension, $d^{(stop)}$, by providing orthogonal roots for the reduced dimensions from d to $d^{(stop)}$. In Figure 4.3 we see the benefit of using such a method to avoid the increased run-time of the numerical solver.

Let $\mathcal{S}^{ORCAS}$ be the sample set and U be the change of basis provided by ORCAS for dimensions d to $d^{(stop)} + 1$ and $\mathcal{S}^{(NS)}$ provided by the numerical solver for the remaining dimensions. Then $S = \mathcal{S}^{ORCAS} \cup \{Us : s \in \mathcal{S}^{(NS)}\}$ we have that the objective function is given by

$$\frac{1}{36}\|S^{-T}p_U(\mathcal{S}^{(NS)})\|^2 + \frac{\sigma^2}{4}\|(S^{(NS)})^{-1}\|_F^2 + \frac{\sigma^2}{4}(d - d^{(stop)}).$$

For $\mathcal{S}^{ORCAS}$ there is no approximation error contribution and the noise error is minimized since the difference vectors are by construction orthogonal roots.

We now state two fundamental properties of the reduced problem. To do so, we rely on the

formalism developed for making a polynomial more sensitive and increasing the region with no roots in Section 4.3.3. Recall, that there we introduced a transform $T_{a,v}$ which increases the sensitivity in the direction v. Furthermore, it increases the cone of no roots. We assume that there is a cone of no roots given by $C_{\beta,v}$.

Now, assume that ORCAS has reduced the problem to a $d^{(stop)}$-dimensional subspace W, with $W^{\perp}$ such that there exist $b_i \in W^{\perp}$ with $b_i \notin C_{\beta,v}$ for $1 \leq i \leq d - d^{(stop)}$. This will be the case because ORCAS up to dimension $d^{(stop)}$ will have found $d - d^{(stop)}$ orthogonal roots. Since $C_{\beta,v}$ has no roots, no element will be in the cone.

Given this, invoking Lemma 70 we have a relation on how close at least one vector has to be to v. This immediately tells us that we will retain some increased sensitivity in the reduced problem. The worst case is when the subspace W directly includes v.

Furthermore, invoking Lemma 71 we have that a cone of no-roots is retained in the subspace. For this, we can use the previous result on the closest vector in W to v to obtain a bound on the size of the cone. Again, in the worst case the full cone may be included.

Overall, if β is large in the original space, the difficulty will generally be retained in the reduced problem. However, no additional difficulty will be introduced and in some cases the problem may become easier.

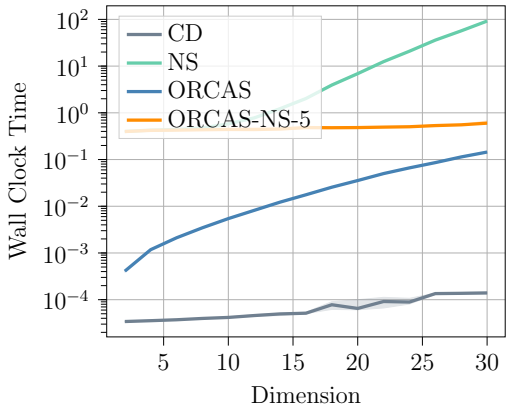


Figure 4.3: Wall Cock Time of Gradient Estimation methods.

4.6 Numerics: Exact Third Order Information

We first study the behavior of the various gradient-estimation methods provided exact third-order information. While unrealistic in practice, it allows us to both properly study the behavior of these methods and observe their best-case performance. To do so, we devise a method to sample random-polynomials allowing us to control various aspects of the problem. Specifically, this allows us to both adjust the impact of lower and higher-order errors and adjust the conditioning of the problem. We then consider the performance of each gradient estimation as a function of this random polynomial.

4.6.1 Sampling random polynomials

We write the random homogeneous polynomial of order k as $P^{(k)}(x)$ with a realization written as $p^{(k)}(x)$. A random polynomial of order k is then simply the sum of all random homogeneous polynomials with order less than or equal to k.

For the experiments we want the random homogeneous polynomial to satisfy the following properties where M_k and a are tunable parameters.

1. The average size of $P^{(k)}$ is M_k, more specifically, $\mathbb{V}_{X,P}[P^{(k)}(X)] = M^2$ with $X \sim \mathcal{N}(0, \frac{1}{\sqrt{d}} I_{d\times d})$.
2. For x and x' such that $\|x\| = \|x'\|$ we have that $P^{(k)}(x) \sim P^{(k)}(x')$ which is equivalent to the polynomial being distributionally invariant to rotations.
3. To make the polynomial ill-conditioned we write $P^{(k)}_{a,v}(x) = P^{(k)}(T_{a,v}(x))$ with v being a unit vector and a a given sensitivity level, where the $T_{a,v}$ transform is as introduced in Definition 65.

The first property relates to the average size of the polynomial. Here we choose the size to be measured by averaging over the domain as weighted by a Gaussian distribution. We choose the variance to scale as $\frac{1}{\sqrt{d}}$ so that the average size of X is one. In conjunction with the second

property we can then reason about the average size of the polynomial a unit distance away from the origin as given by the following result.

Lemma 79. *Let $X \sim \mathcal{N}(0, \frac{1}{\sqrt{d}} I_{d\times d})$ and p be a random polynomial of order k such that $P(x) \sim P(x')$ if $\|x\| = \|x'\|$. Then for any unit vector $\hat{x}$ we have*

$$\mathbb{V}_P[P(\hat{x})] = \mathbb{V}_{X,P}[P(X)] \frac{d^k}{d(d+2)\cdots(d+2(k-1))}.$$

The proof is given in Section 4.8 4.8.2. The above result tells us that the size of the polynomial a unit distance from the origin is approximately $\sqrt{\mathbb{V}_P[P(\hat{x})]}$, becoming tighter as the dimension increases.

When the condition number a is one, then the average size of the polynomial is simply given by M. To reason about the size when the condition number is not equal to one, we present the following result using the third property. Since we will only vary the condition number for the third order homogeneous polynomial, we present the result only for such cases. The proof can be extended to more general dimensions if needed.

Lemma 80. *Let p be a homogeneous polynomial of order 3 and let it have the property that $P(X) \sim P(x')$ where $\|x\| = \|x'\|$ then we have*

$$\begin{aligned}\mathbb{V}_{X,P}[P_{a,V}(X)] &= \frac{\mathbb{V}_{X,P}[P(X)]}{d(d+2)(d+4)}\left(15a^6 + 9a^4(d-1) + 3a^2(d-1)(d+1) + (d-1)(d+1)(d+3)\right) \\ &\leq M^2\left(15\left(\frac{a^2}{d}\right)^3 + 9\left(\frac{a^2}{d}\right)^2 + 3\left(\frac{a^2}{d}\right) + 1\right).\end{aligned}$$

The proof is given in Section 4.8 4.8.2. We observe that if the condition number a is fixed, that $\mathbb{V}_{X,P}[P_{a,V}(X)]$ will approach the variance if the condition number is one as the dimension is increased. For a fixed dimension, the size of the polynomial is dominated by the cube of a. Observe that to keep the size constant with the dimension, a has to grow proportional to $\sqrt{d}$.

First Property: Sample Homogeneous Polynomials of Average Size M

The randomness of the polynomial arises from the choice of coefficients c. To make this explicit, we write $p(x;c) = \sum_{|\alpha|=k} c_\alpha x^\alpha$. We denote the upper-case letter C to be the random coefficients.

We choose $C_\alpha \sim \mathcal{N}(0, \sigma_\alpha)$ and find σ_α such that for $X \sim \mathcal{N}(0, \frac{1}{\sqrt{d}} I_{d\times d})$ and $\alpha \neq \alpha'$ we have $\mathbb{V}_{X,P}[C_\alpha x^\alpha] = \mathbb{V}_{X,P}[C_{\alpha'} x^{\alpha'}]$. So, each term contributes the same amount on average. This is to ensure symmetry in the random polynomial and hence no direction will be more sensitive than all others. Due to the symmetry, heuristically orthogonal roots may be more likely as we confirm in the Numerics.

Lemma 81. *Let $C_\alpha \sim \mathcal{N}(0, \sigma_\alpha^2)$ where $\sigma_\alpha = M\sqrt{\frac{d^k}{\binom{d+k-1}{d-1}\prod_{i,\alpha_i>0}(2\alpha_i-1)!!}}$ and $X \sim \mathcal{N}(0, \frac{1}{d} I_{d\times d})$ we have $\mathbb{V}_{X,P}[P^{(k)}(X)] = M^2$.*

Proof. Using the law of total variance we have that

$$\mathbb{V}_{X,P}[P^{(k)}(X)] = \mathbb{E}_X[\mathbb{V}_P[P^{(k)}(X)]] + \mathbb{V}_X[\mathbb{E}_P[P^{(k)}(X)]].$$

Since each C_α is mean zero we have that $\mathbb{E}_P[P^{(k)}(x)] = 0$. Due to C_α and $C_{\alpha'}$ being independent when $\alpha \neq \alpha'$ and x being given we have that $\mathbb{V}_P[P^{(k)}(X)] = \sum_{|\alpha|=k} x^{2\alpha}\mathbb{V}_P[C_\alpha]$. Taking the expectation with respect to X we have

$$\begin{aligned}\mathbb{V}_{X,P}[P^{(k)}(X)] &= \sum_\alpha \mathbb{V}_P[C_\alpha]\mathbb{E}_X[X^{2\alpha}] \\ &= \sum_\alpha \sigma_\alpha^2 \frac{1}{d^k} \prod_{i,\alpha_i>0} (2\alpha_i - 1)!!.\end{aligned}$$

where $\mathbb{E}_X[x^{2\alpha}] = \frac{1}{d^k}\prod_{i,\alpha_i>0}(2\alpha_i-1)!!$ is due to independence and properties of Gaussian moments.

Notice that there are $\binom{d+k-1}{d-1}$ monomial elements in the homogeneous polynomial of order k. The result immediately follows. □

Remark 7. *Since the number of coefficients grows exponentially as we increase the order of the polynomial, we also allow for sub-sampling of the coefficients. Letting N_{sub} be the number of*

coefficients sampled, we simply multiply the resulting polynomial by $\sqrt{\frac{\binom{d+k-1}{d-1}}{N_{sub}}}$. *This results in a polynomial with identical properties to a polynomial using all coefficients.*

Second Property: Rotational Invariance

As it stands, the random polynomial is not yet guaranteed to be distributionally invariant to rotations. For this we would need $P^{(k)}(x) \sim P^{(k)}(x')$ when $\|x\| = \|x'\|$.

To accomplish this, let U be a random rotation matrix, such that for any x we have that Ux is uniformly distributed among all vectors with norm given by $\|x\|$. We use the scipy `special_ortho_group` function to accomplish this [VGO+20].

Consider now the random polynomial $P^{(k)}(Ux)$. To see this transform leads to the random polynomial being distributionally invariant simply note that the input to $P^{(k)}$ is uniformly random overall all vectors with the same norm. From now on, we let the random polynomial function be given by $P^{(k)}(x) := P^{(k)}(Ux)$.

Third Property: Control Conditioning

The last property is easily satisfied. We simply write $P^{(k)}_{a,v}(x) = P^{(k)}(T_{a,v}(x))$. If we let v be a random vector V which is uniformly random among all unit vectors, then it easy to see that the resulting polynomial remains distributionally invariant to rotations.

While this transform increases the sensitivity in a given direction, we cannot guarantee that there do not exist orthogonal roots. Using Proposition 74 we can only state the existence of an a such that with some non-zero probability the polynomial will not have a set of d orthogonal roots.

It is not trivial to decide which a is appropriate. Proposition 68 tells us that if the size of the region with no-roots does not change with dimension, we have to increase $a \propto \sqrt{d}$. However, it may be the case that regions with no-roots also grow with dimension. Lemma 80 indicates furthermore, that the induced sensitivity due to the transform diminishes with the dimension.

To keep the sensitivity constant a would have to again grow as $\sqrt{d}$. In the experiments we both present the results when we keep a constant and when we grow it as $\sqrt{d}$.

4.6.2 General Methodology

We can now use the random polynomials to assess the performance of various gradient estimation methods. We will only consider polynomials of fifth order. We do so because that is the first error-order which we do not explicitly optimize for. The benefit of the current set-up is that we can express the mean squared error exactly and use all the previous results to reason about the experiments. We will specifically lean on the case study presented in Section 4.2.4. Throughout the set-up we write $P^{(k)}$ for a random polynomial and do not yet distinguish between a polynomial which has or has not been ill-conditioned.

Note that for a given instance of a fifth-order polynomial have the expression

$$\begin{aligned}\mathbb{E}_\epsilon[\|\nabla_S^{(R)}\tilde{p}(\mathbf{0}) - \nabla p(\mathbf{0})\|^2] = \|S^{-T}p^{(3)}(\mathcal{S})\|^2 + \frac{\sigma^2}{4}\|S^{-1}\|_F^2 \\ + \langle S^{-T}p^{(3)}(\mathcal{S}), S^{-T}p^{(5)}(\mathcal{S})\rangle + \|S^{-T}p^{(5)}(\mathcal{S})\|^2\end{aligned} \tag{4.6.1}$$

We chose $x_0 = \mathbf{0}$ as then we have that $D^{\otimes k}p(x_0)\|_{x_0=\mathbf{0}}[s] = k!\ p^{(k)}(s)$, leading to the above simplified mean squared error.

We are now interested in the performance of gradient estimation methods over the distribution of polynomials. Notably, here the optimal sample set is then also a random variable as it is becomes a function of P. We denote this random variable by S_P. We then want to compare methods via the summary statistic $\mathbb{E}_{P,\epsilon}[\|\nabla_{S_P}^{(R)}\tilde{P}(\mathbf{0}) - \nabla P(\mathbf{0})\|^2]$.

While the above description is reminiscent of the description in the case study in Section 4.2.4, by taking the average over the polynomials, we lose some interpretability. This is due to the lack of a closed formed solution due to the difficult to interpret distribution of S_P. Instead, we use a Monte Carlo estimator of the above summary statistic.

While the summary statistic is difficult to immediately interpret, we can structure the experiments

in a way to reason about the possible best-case performance of each method. The following will then rely on results from the case study, specifically Section 4.2.4.

Comparison to Baseline

We now want to compare the methods in various regimes based on the best-case performance attainable. The issue remains that due to taking an average of the distribution of polynomials, we cannot easily construct a baseline, since it will itself be a function of the random polynomial. Instead, we look at a baseline based to an average polynomial. In other words, we want to find a baseline S independent of P which minimizes the mean squared on average.

For the baseline we choose $S = rU$ where U is an orthogonal matrix and $r > 0$ will depend on the problem parameters. The choice of an orthogonal matrix is due to an orthogonal matrix a-priori minimizing the noise error as given in Lemma 64. We then have that the mean squared error simplifies to

$$\mathbb{E}_{P,\epsilon}[\|\nabla_S^{(R)}\tilde{P}(\mathbf{0}) - \nabla P(\mathbf{0})\|^2] = r^4\mathbb{E}_P[\|P^{(3)}(\mathcal{U})\|^2] + r^8\mathbb{E}_P[P^{(5)}(\mathcal{U})\|^2] + \frac{\sigma^2}{4}\frac{1}{r^2}$$

The cross-term disappeared due to the coefficients of the polynomials being independent and mean zero.

Now, observe that $\mathbb{E}_P[\|P^{(k)}(\mathcal{U})\|^2] = \sum_{i=1}^d \mathbb{V}_P[P^{(k)}(u_i)]$. Since u_i are all unit-vectors, we can directly invoke Lemma 79 telling us that $\mathbb{V}_P[P^{(k)}(\hat{x})]$ is the same for all unit vectors $\hat{x}$. Hence, we can write $\mathbb{E}_P[\|P^{(k)}(\mathcal{U})\|^2] = d\mathbb{V}_P[P^{(k)}(\hat{x})]$ where $\hat{x}$ is any unit-vector. Hence, we can significantly simplify the mean squared error for the baseline

$$\mathbb{E}_{P,\epsilon}[\|\nabla_{rU}^{(R)}\tilde{P}(\mathbf{0}) - \nabla P(\mathbf{0})\|^2] = r^4 d\ \mathbb{V}_P[P^{(3)}(\hat{x})] + r^8 d\ \mathbb{V}_P[P^{(5)}(\hat{x})]\ + \frac{\sigma^2}{4}\frac{d}{r^2} := \ell_{P,\sigma}^{(Base)}(r).$$

We now define the two central quantities to assess what the best baseline improvement may be. To obtain a more in-depth discussion see the case study, specifically Section 4.2.4. We have that $h = \left(\frac{\sigma^2}{16\ \mathbb{V}_P[P^{(5)}(\hat{x})]}\right)^{1/10}$ and $\tilde{h} = \left(\frac{\sigma^2}{8\ \mathbb{V}_P[P^{(3)}(\hat{x})]}\right)^{1/10}$. Here h represents the optimal radius when

there is no third-order error and $\tilde{h}$ represents the error when the fifth order error is zero. We have shown in the case-study section that the minimum of the two closely approximates the true optimal radius of the above error.

We compare the loss achieved by the baseline to a loss given by

$$\underline{\ell}_{P,\sigma}^{(Base)}(r) := r^8 d \, \mathbb{V}_P[P^{(5)}(\hat{x})] \; + \frac{\sigma^2}{4}\frac{d}{r^2}.$$

We argue that this is very close to the best case a method which does not explicitly optimize for the fifth-order error can achieve. Here the noise error has been minimized and the third-order error fully removed. The r which minimizes this will then be given by h, by definition of h. The true error may be slightly off, but we hope that on average the fifth-order contribution is $\mathbb{V}_P[P^{(5)}(\hat{x})]$ which would then yield the above.

Since the result is equivalently derived we restate the result from Section 4.2.4 to conform to the notation used here.

Lemma 82 (Restated Lemma 60)**.** *Assume that $\tilde{h} \leq h$ and $\tilde{h} = \gamma h$ then $\frac{\ell_{P,\sigma}^{(Base)}(r)(\tilde{h})}{\underline{\ell}_{P,\sigma}^{(Base)}(r)(h)} = \mathcal{O}\left(\frac{1}{\gamma^2}\right)$. Now, when $\tilde{h} \geq h$ and $h_{\hat{S}} = \gamma \tilde{h}_{\hat{S}}$ then we have $\left|\frac{\ell_{P,\sigma}^{(Base)}(r)(h)}{\underline{\ell}_{P,\sigma}^{(Base)}(r)(h)} - 1\right| = \mathcal{O}(\gamma^6)$.*

Objective Function Performance

In conjugation to studying the performance on the full gradient estimation problem, we also consider the performance of the method based on solving the problem designed to solve, that is minimizing the objective function presented in Definition 54. Of course, we here need to choose a fixed parameter h to make the problem well-defined. The h chosen will be the one presented above. That is, it will be the best-case improvement under only fifth-order errors. As argued in the case study, specifically Section 4.2.4, the objective function loss should mimic the gradient estimation performance already. Specifically, in a setting where we do not incorporate the structure of the higher-order errors.

For the objective function we can use the results we have found for the worst-case performance

in Section 4.4.5, where the central difference case reduced to having the form rU just the above baseline. Here we then have the following. Recall that $\ell^{(lb)}_{p,\sigma,h}$ is the best case error for a fixed upper-bound h.

$$\frac{\ell^{(lb)}_{p,\sigma,h}}{\ell_{p,\sigma,h}(h^*_{\hat{S}}U)} = \frac{1}{2}(h^*_{\hat{S}})^4\frac{h^2}{\tilde{h}^6} + \frac{h^2}{(h^*_{\hat{S}})^2}.$$

Similar to the gradient estimation case, we again have that $h \geq \tilde{h}$ that $\frac{\ell^{(lb)}_{p,\sigma,h}}{\ell_{p,\sigma,h}(h^*_{\hat{S}}U)} = \mathcal{O}(\frac{h^2}{\tilde{h}^2})$. Otherwise, the error behaves as $\left|\frac{\ell^{(lb)}_{p,\sigma,h}}{\ell_{p,\sigma,h}(h^*_{\hat{S}}U)} - 1\right| = \mathcal{O}(\frac{h}{\tilde{h}}^6)$. So the same regimes occur.

Set-up

To study the gradient estimation problem systematically, we recall that there are five parameters given by $M^{(3)}, M^{(5)}, a, \sigma, d$, which fully specify the problem. We will discuss each parameter and discuss how we vary them in the following. At the end, we discuss the hyperparameter in the objective function in Definition 54, which we denote by h^{HP} to distinguish it from the h derived directly from the problem properties.

The ill-conditioning level a fundamentally alters the properties of the problem. Hence, the experiments will be split in two sections, each investigating a different regime. The first regime will focus on the well-conditioned case, that is $a = 1$. The latter will consider the ill-conditioned case. For the remainder of the set-up, it will suffice to assume that we have fixed an a.

We do this in various dimensions, as many results, such as the worst-case performance of ORCAS presented in Section 4.4.5, rely on the dimensionality. Specifically, we consider all even dimensions from two to twenty. For each dimension, we sample 100 polynomials and compute the mean squared error. The mean squared error can be analytically computed using Equation 4.6.1 given the sample set provided by a gradient estimation method. Hence, the variation which we will be plotting solely arises from the differences in performance on sampled polynomials. In the plots, we plot the median performance with error-bands representing the 25th and 75th percentiles.

Next, we discuss how for a given a and d we investigate the regimes given by the relative sizes of h and $\tilde{h}$. We fix h and then will vary $\tilde{h}$ to achieve the various regimes. The guiding principle will be to have a regime $\tilde{h} \ll h$ where we expect large improvement gains. Then a regime where $\tilde{h} \approx h$ to investigate the potential gain in the borderline case. Lastly, we consider $\tilde{h} \gg h$ to confirm that there is little gain in removing the third-order error.

To understand how to adjust h and $\tilde{h}$, notice that each depends on $\mathbb{V}[P^{(k)}(\hat{x})]$, where k provides the corresponding order. Given Lemma 79 we have that $\mathbb{V}[P^{(k)}(\hat{x})]$ is a function of both the average size of the polynomial given by $\mathbb{V}_{P,X}[P^{(k)}(X)]$ and the dimension d. The result states that for large dimensions the size of the polynomial at a unit-vector is equal to the average size of the polynomial. In the case when the polynomial is well-conditioned, that is $a = 1$, the average size of the polynomial is directly given by $M^{(k)}$. Otherwise, we use Lemma 80.

Since we only vary a for the third-order homogeneous polynomial, the average size of the fifth-order homogeneous polynomial is given by $M^{(k)}$. Hence, we have using Lemma 79 that

$$h = \left(\frac{\sigma^2}{16(M^{(5)})^2} \frac{d(d+2)\cdots(d+8)}{d^5} \right)^{1/10}.$$

We make the choice to not vary $M^{(5)}$ as we vary the dimension. Hence, h will vary with the dimension also. The choice is based on the fact that we interpret $M^{(5)}$ as a fundamental property of the polynomial and would like to keep it fixed.

For the quantity $\tilde{h}$ we can perform a similar analysis, which will simply be complicated by the condition number in a. Using Lemma 80 we will address this in each of the Sections where we vary the condition number.

We now discuss which parameters to choose to accomplish different relative sizes of h and $\tilde{h}$. Notice here that σ is arbitrary, since we can simply scale $P^{(k)}$ by a constant so that via $\mathbb{V}_P[P^{(k)}(\hat{x})]$ the noise is adjusted. However, since $\sigma = 0.001$ is a common noise-level, we choose the same here. We then also choose $M^{(k)} = 150$ to achieve $h \approx 0.07$ for large dimensions. It now remains to choose $M^{(3)}$ to place the problem into the relevant regimes. Since the dimension slightly changes both the size of h and $\tilde{h}$, we cannot have the ratio be exactly equal for each

dimension. We will aim that for $\tilde{h} \ll h$ that $\frac{\tilde{h}}{h} \approx 0.1$, for $\tilde{h} \approx h$ we have $\frac{\tilde{h}}{h} \approx 1$ and for $\tilde{h} \gg h$ we have $\frac{\tilde{h}}{h} \approx 10$.

To accommodate for the varying h with dimension and for the fact that h is unlikely to be the optimal radius for samples from the random polynomial, we select a range of values for the hyperparameter h^{HP}. We have that in the range of dimensions we consider that $0.07 \leq h \leq 0.11$. Hence, we choose the hyperparameter accordingly, with a bit of extra room below 0.7 to allow for variance in the samples from the random polynomials.

Here we provide a summary of the above.

- We split the problem into two parts depending on the conditioning of the polynomial.
- We consider all even dimension from two to 20.
- We fix $\sigma = 0.001$ and $M^{(5)} = 150$ which yields $h \approx 0.07$.
- We consider three performance regimes and will adjust $M^{(3)}$ accordingly. Specifically for $\tilde{h} \ll h$ that $\frac{\tilde{h}}{h} \approx 0.1$, for $\tilde{h} \approx h$ we have $\frac{\tilde{h}}{h} \approx 1$ and for $\tilde{h} \gg h$ we have $\frac{\tilde{h}}{h} \approx 10$.
- Since h varies with dimension we choose the corresponding hyperparameter in the objective function to be selected from $h^{\text{HP}} \in [0.11, 0.1, 0.09, 0.08, 0.07, 0.06]$. We then select the best hyperparameter for every sample.
- For the objective function plots we use $h^{\text{HP}} = 0.08$ as this is what h evaluates to for most dimensions.
- For the MC estimate we sample $N = 100$ random polynomials. We then plot the median and the 25-75 percentile.
- Subsample coefficients, where we let $N_{sub} = d^3$. So, of the same order as the degree three polynomials.

Gradient Estimation Methods

We are considering various gradient estimation methods. We summarize them in the list below.

- Numerical Solver (`NS`): We use the numerical solver as described in Section 4.5.
- Central Differences (`CD`): We use an orthogonal set with the difference vectors adjusted optimally as given in Proposition 55.
- `ORCAS`: We described the method in Section 4.4.2 and is used for the complete method.
- `ORCAS-NS-5` and `ORCAS-NS-8`: We use ORCAS as a dimensionality reduction method to then use the numerical solver on a smaller problem. We describe this methodology in Section 4.5. We use two different stopping dimensions to assess the difference in performance.
- Lower and Upper Bound: Uses the bounds as provided by the baseline methods in Section 4.6.2 and 4.6.2. Recall, besides the lower-bound on the objective function, they are not strict bounds as they are derived only for a baseline method. However, we would expect them to closely match the true lower and upper bounds on performance.

4.6.3 Well-Conditioned Polynomials

In the case when there is equal conditioning, that is $a = 1$, we recall that by construction $\mathbb{V}_{X,P}[P^{(k)}(X)] = M^{(k)}$. Combined with Lemma 79 we have that

$$\tilde{h} = \left(\frac{\sigma^2}{8(M^{(3)})^2} \frac{d(d+2)(d+4)}{d^3} \right)^{1/6}.$$

To achieve the appropriate ratios of h and $\tilde{h}$, we simply let $M^{(3)} \in [0.01, 1, 100]$.

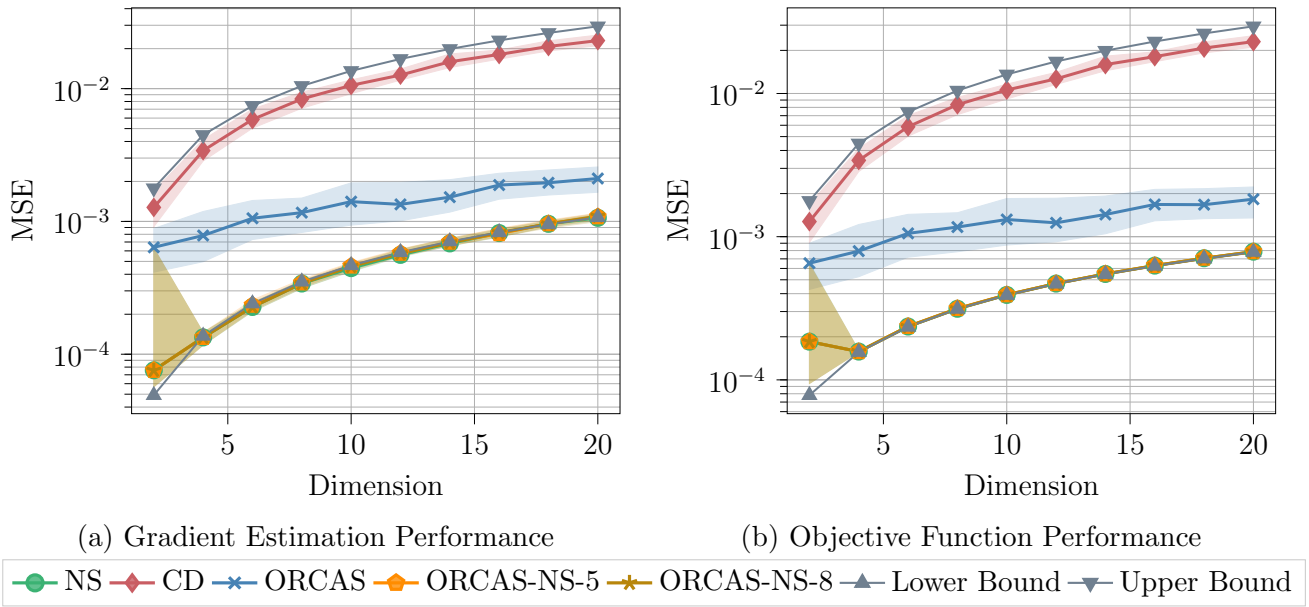
Performance Regime: $\tilde{h} \ll h$

In Figure 4.4 we observe an order of magnitude improvement of all newly developed methods compared to the baseline and the standard central difference method–justifying the development of such methods.

In Figure 4.4a, the methods based on the numerical solver reach the best-case performance by the fourth dimension. `ORCAS` approaches this performance with increasing dimension as was predicted by the results in Section 4.4.5 discussing its worst case performance. Notably, the lower-bound performance is not a strict lower-bound on the performance, it only indicates the best-case performance of a method which fully removes the third order error while being agnostic to the structure of higher-order errors.

In Figure 4.4b the lower-bound is a strict lower-bound. The methods that achieve this lower bound provide optimal solutions to the optimization problem given in Definition 54. Furthermore, by Proposition 62 solutions which achieve this lower bound can only be given by sets of d orthogonal roots. Due to numerical inaccuracies, it may be that these solutions are not exactly orthogonal roots; however, they appear to be very close.

Lastly, an important observation is that the dimensionality reduction methods perform equivalent to the purely numerical method. Indicating that in such situations these are advantageous, especially in even higher dimensions.

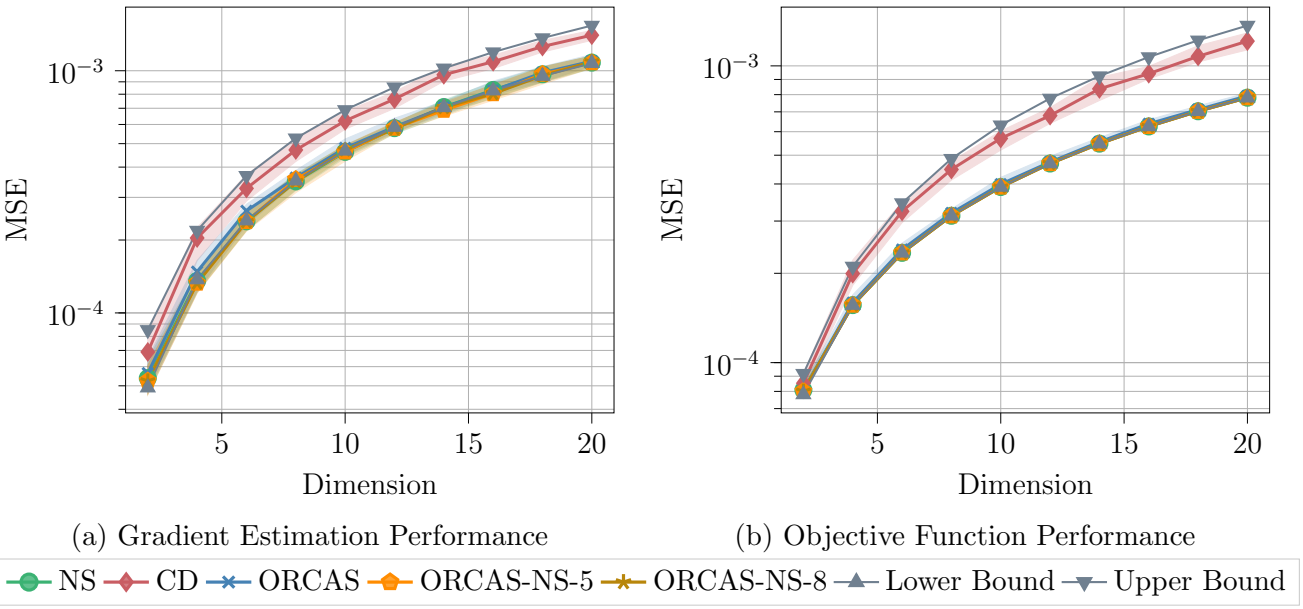


(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.4: Mean squared error as a function of dimension for the performance for well-conditioned polynomials in the performance regime $\tilde{h} \ll h$.

Performance Regime: $\tilde{h} \approx h$

In Figure 4.5 we observe the performance difference be smaller as now the higher and lower order approximation errors contribute similarly. `ORCAS` in this case is very close to optimal performance also. Central differences retains its poor performance.

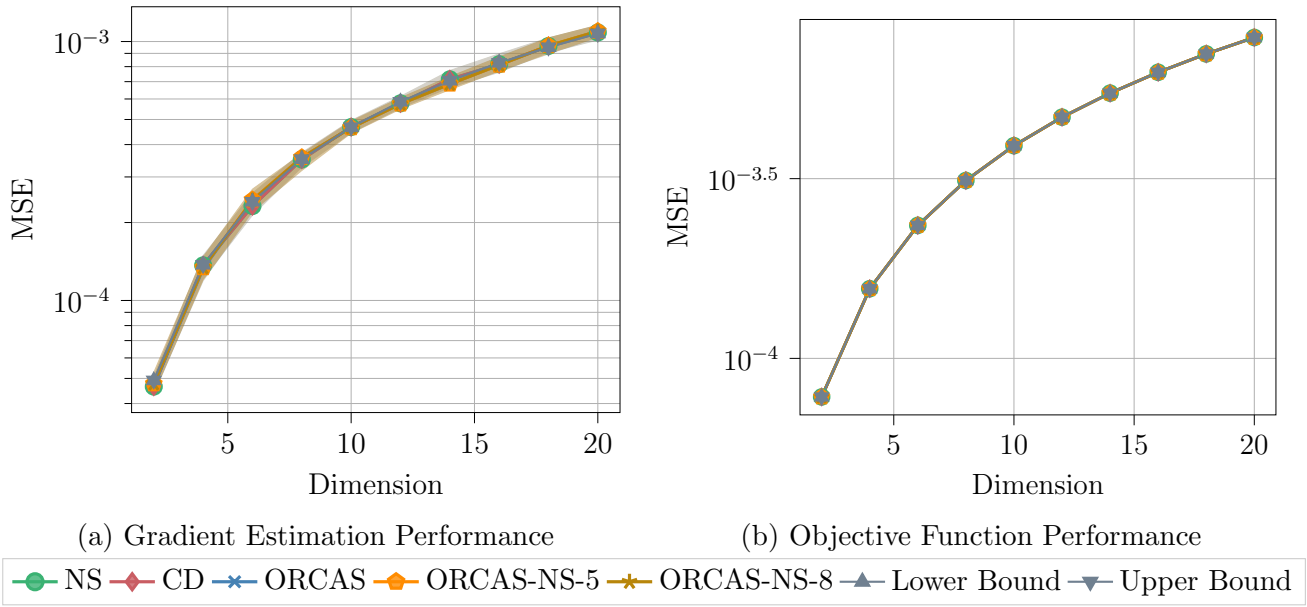


(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.5: Mean squared error as a function of dimension for the performance for well-conditioned polynomials in the performance regime $\tilde{h} \approx h$.

Performance Regime: $\tilde{h} \gg h$

In Figure 4.6 we observe no more performance difference between the methods. The higher-order errors are dominating and hence reducing the lower-order errors has little effect. The optimal method is simply an orthogonal sample set, as reducing the noise error has the biggest impact on performance.



(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.6: Mean squared error as a function of dimension for the performance for well-conditioned polynomials in the performance regime $\tilde{h} \gg h$.

4.6.4 Ill-Conditioned Polynomials

Increasing the condition number a has two effects. For one, it increases the sensitivity of the polynomial in a random direction. This hinders the performance of gradient estimation methods as in such directions difference vectors cannot be made as long to mitigate the noise-error. The other effect is that an increased condition number also increases the region of no roots. By Proposition 72, when the condition number is large enough the polynomial will not have a set of d orthogonal roots. With this, a gradient estimation method will have to make a trade-off between sample points that are roots and ones which conform to orthogonality.

The effect of the condition number differs depending on the dimension. Hence, we consider two separate experiments. One focuses on keeping the condition number constant while the other increases the condition number proportional to $\sqrt{d}$.

Keeping the condition number constant retains the same sensitivity in a given direction. However, as the dimension increases, the size of the polynomial decreases as is given by Lemma 80. This happens because the relative volume of the sensitive region decreases with an increasing dimension. Based on the Lemma, for large enough d the size of the polynomial will remain

constant when increasing the condition number proportional to $\sqrt{d}$. Justifying the strategies for varying a.

Discussing the impact of the condition number on the existence of orthogonal roots is more heuristic. For this, recall that an increase in the condition number increases a region with no roots, as we have discussed in Section 4.3.3. When the region is enlarged enough there cannot be any orthogonal roots via Proposition 68. The difficulty is that it is not a-priori clear how large the original region of no-roots is and hence how vary the condition number.

While by trial and error we may find a condition number that appears to ensure that no-orthogonal roots exist, it remains unclear how to adjust this number as the dimension increases. We know by Proposition 68 that if the region of no-roots remains constant we need to increase a proportional to $\sqrt{d}$. However, it may be that as we increase the dimension, the size of regions without roots increase. This case is then addressed by experiments keeping the condition number constant, as here the regions without root may not have to be further increased.

To better discuss the results, we note a few properties of the gradient estimation methods in relation to the increased condition number.

Methods which enforce an orthogonal sample set, such as ORCAS and CD, will suffer more from increased condition numbers. This is because one sample point is likely to be significantly along the sensitive direction and hence need to be shorter and unable to offset the noise error. For ORCAS, Lemma 70 makes this concrete, as it provides the relation that with an increased condition number the final sample point will be increasingly close to the sensitive direction. The benefit of a numerical solver is then apparent, as it can better avoid such sensitive regions. However, this will sacrifice orthogonality and potentially overfit. A trade-off we discuss in Section 4.7.

Furthermore, we recall the properties of ORCAS as a dimensionality reduction method. Note from Section 4.5.2 that when using `ORCAS-NS`, the problem solved by the numerical solver at dimension $d^{(stop)}$, will likely have similar properties as the original d dimensional problem. Specifically, the sensitivity and the size of the region without roots in the lower-dimensional space

will be in terms of the same properties of the original polynomial. Hence, we may intuitively think of the $d^{(stop)}$ dimensional problem as having the same difficulty as the base problem. When a is proportional to $\sqrt{d}$ this is important, as a problem that had been reduced from a d to a $d^{(stop)}$-dimensional problem will be more difficult to solve than if the base problem had been $d^{(stop)}$ dimensional.

4.6.5 Fixed condition number

We first fix $a = 10$ and keep it constant as we increase the dimension. We found this value to be a good trade-off in difficulty and solve-ability. In the random direction v the sensitivity will then be increased by 10^3. Since the average size of the polynomial is increased, we choose $M^{(3)} \in \{0.35, 0.0035, 0.000035\}$, which for a polynomial of dimension four provides the same average polynomial size as earlier. Specifically, this will yield $\frac{\tilde{h}}{h} \in \{0.1, 1, 10\}$.

Notably, the average size of the polynomial decreases with dimension as seen by Lemma 80. This leads $\tilde{h}$ to increase. Therefore, if $\tilde{h} \leq h$ then as the dimension increases it will get closer to h, making the potential performance gain smaller. However, it will also allow for potentially more separation between CD and ORCAS as the sensitive region will matter relatively less.

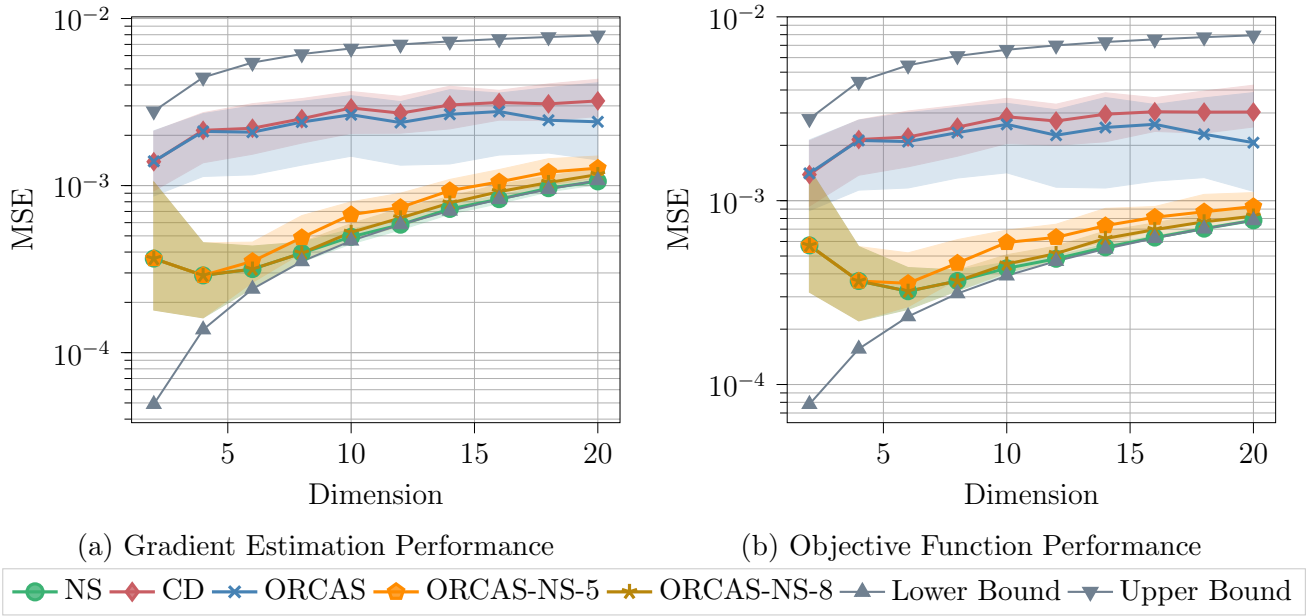
Performance Regime: $\tilde{h} \ll h$

In Figure 4.7 we again observe an order of magnitude improvement compared to baseline performance. However, as suggested above, we confirm that the performance gap narrows as the dimension increases. This is due to $\tilde{h}$ increasing based on the size of the third order polynomial decreasing as proved in Lemma 80.

We also observe that while at first `ORCAS` performs similarly to `CD`, as the dimension increases they seperate in performance and `ORCAS` approaches the lower bound. This is due to the sensitivity impacting the models relatively more for lower dimensions. As the dimension increases, the one sensitive direction will matter relatively less and hence the performance gap between `ORCAS` and `CD` becomes more apparent.

While the numerical solver based methods struggle attaining the lower bound at first, they also improve as the dimension increases. While the methods can adjust the sample set to avoid the sensitive regions, they can not hit the lower-bound unless they find a set of d orthogonal roots. They appear to do so as the dimension increases as seen in Figure 4.4b. This may indicate that the methods are either not able to find orthogonal roots or that they do not exist anymore for these lower dimensions.

Lastly, we note the separation between the numerical solver based methods depending on the stopping dimension $d^{(stop)}$. As observed above and in Section 4.5.2, the reduced problem's characteristics will be in terms of the characteristics of the original problem. As we fixed the condition number, the characteristics are the same for all problems. This allows us to argue that even for higher-dimensions, the performance of `ORCAS-NS` will be similar to the performance of the numerical solver when solving a base problem of dimension $d^{(stop)}$. A caveat being that as the dimension increases, the difficulty of the reduced problem may reduce by chance, which is also why with increasing dimension many of the methods start improving compared to the lower-bound.



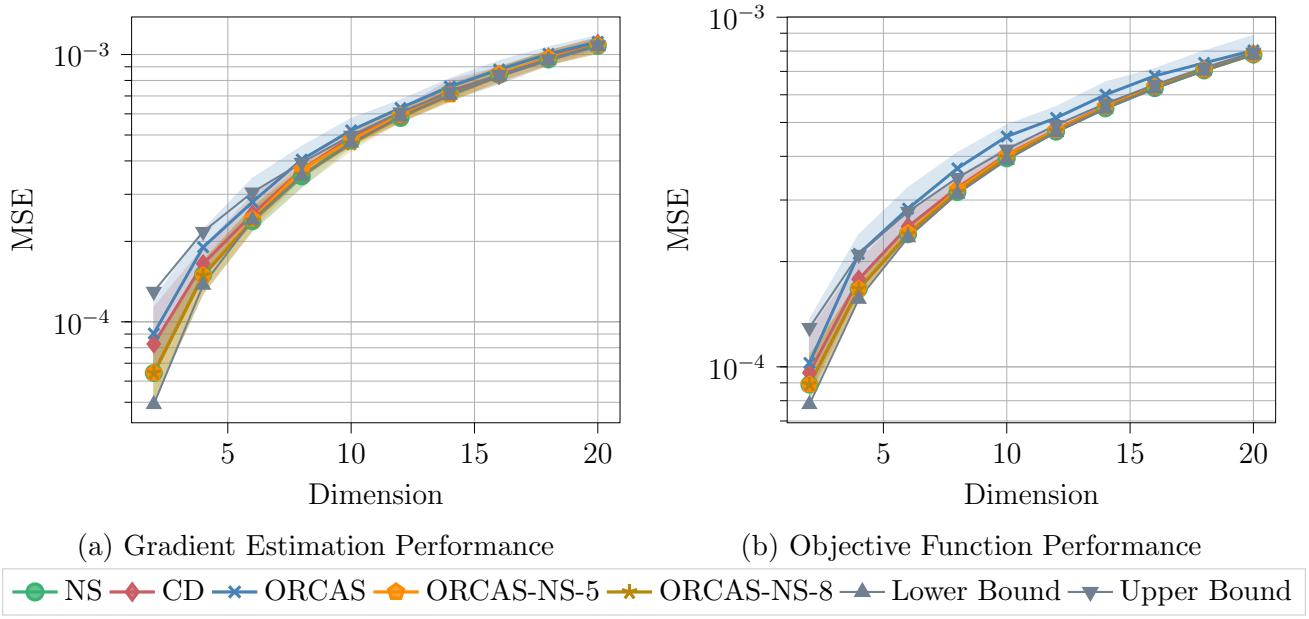
(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.7: Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with a fixed condition number and in the performance regime $\tilde{h} \ll h$.

Performance Regime: $\tilde{h} \approx h$

In Figure 4.8 we observe the performance of all methods to be very similar. The stand-out observation is that `ORCAS` is performing slightly worse than all the other methods. This is because `ORCAS` will force the last orthogonal vector to be very close to the sensitive direction by Lemma 70. This is due to it selecting only roots, which all lie in the complement of the sensitive region. Hence, the last sample point will be close to the sensitive region.

This was not observed in the previous plot, as the benefit of choosing roots for the other dimensions was outweighed by the deficit of being slightly more along the sensitive region than `CD`. However, when modifying the third order error contributions matters little, this trade-off is not worth it anymore.

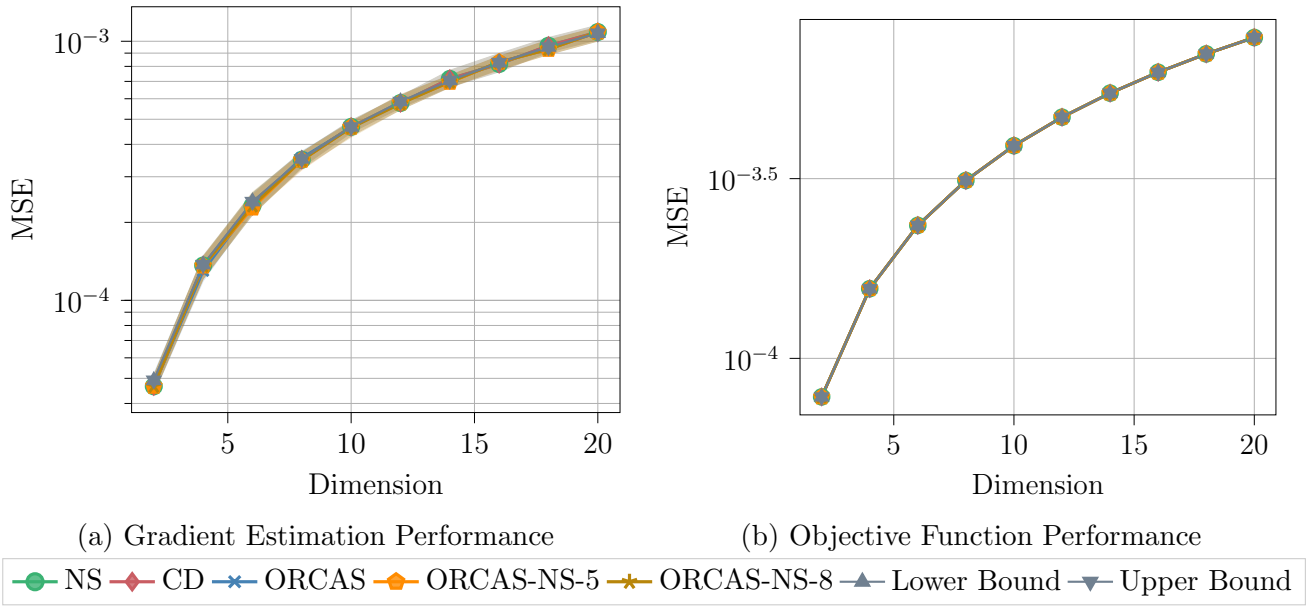


(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.8: Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with a fixed condition number and in the performance regime $\tilde{h} \approx h$.

Performance Regime: $\tilde{h} \gg h$

We see in Figure 4.9 no performance difference between the methods. The higher-order errors are dominating and hence reducing the lower-order errors has little effect.



(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.9: Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with a fixed condition number and in the performance regime $\tilde{h} \gg h$.

4.6.6 Growing Condition Number with Dimension

In this section we let $a = 5\sqrt{d}$ and let $M^{(3)} \in \{0.35, 0.0035, 0.000035\}$ as before. We choose the relation $a = 5\sqrt{d}$ as then for a four dimensional problem the polynomial properties will be identical to the previous experiments where we fixed the condition number.

We note that increasing the condition number with the dimension will also lead to the average size of the polynomial changing, simply due to the size itself depends on the dimension as seen in Lemma 80. However, we have found the increase to be minimal, and therefore $\mathbb{V}_P[P^3(\hat{x})]$ and $\tilde{h}$ can be considered roughly constant.

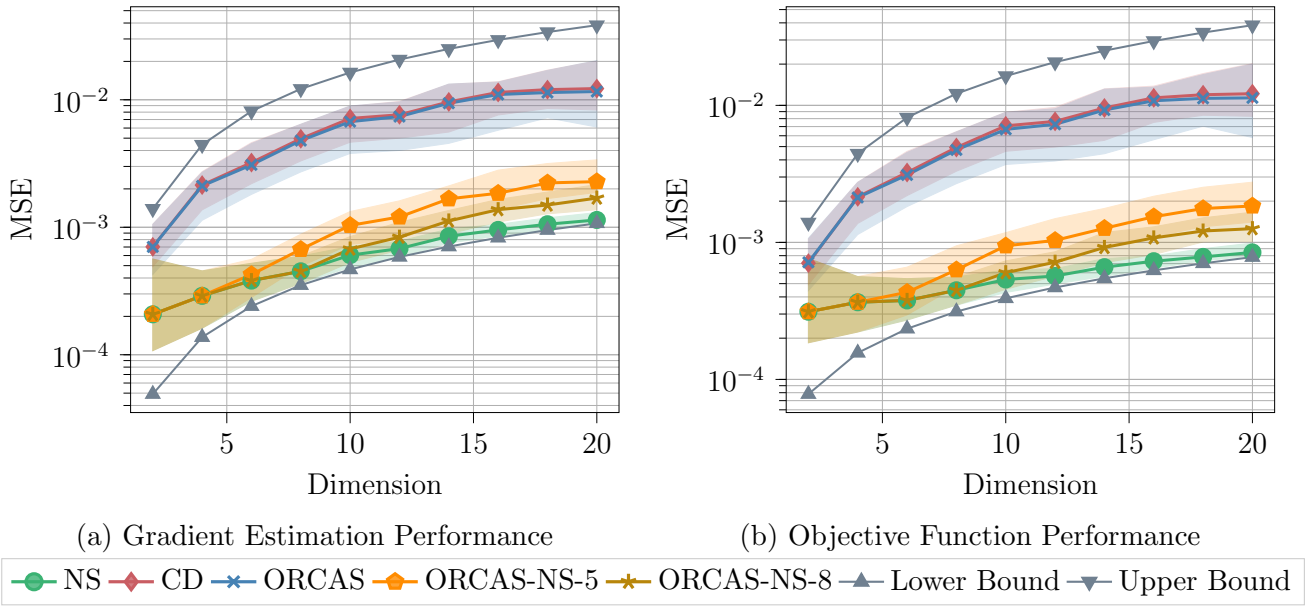
Performance Regime: $\tilde{h} \ll h$

In Figure 4.10 we again see a potential performance improvement of an order of magnitude. Unlike the previous case where we fixed the condition number, here the performance gap remains. By increasing the condition number the size of the polynomial is retained and hence $\tilde{h}$ also remains constant. Leading to the performance gap remaining constant.

The first observation is that ORCAS does not improve on CD anymore. This behavior follows directly from the increasing condition number. Intuitively, the sensitive region retains its impact as the dimension is increased. Since both methods will necessary have a sample point along the sensitive region, this impact will dominate any other improvement the methods make in the sample set selection.

We also observe that the numerical methods need more dimensions to obtain performance close to the lower-bound. Hence, increasing the condition number further increases the difficulty of solving the gradient estimation problem. Recall that this difficulty stems from an increased condition number leading to larger regions with no roots. Now, looking at Figure 4.10b we see that NS is still slowly approaching the lower bound. This means that it is able to find solutions that are compromised of sample points that are close to being orthogonal roots. This can only happen when the condition number was not large enough to make the region of no roots large enough. Then, even as the condition number is increased, it may be that large regions of no-roots become increasingly unlikely with higher-dimensions.

The last observation focuses on the performance of the ORCAS-NS methods. Just like in the case where the condition number is fixed, the methods depending on different stopping dimensions $d^{(stop)}$ separate from the numerical solver. The difference is that now the difficulty of the solving the reduced problem may increase as the condition number of the base problem increases. This explains why the ORCAS-NS methods do not approach the lower-bound as quickly as in the fixed condition number case and may even become worse.

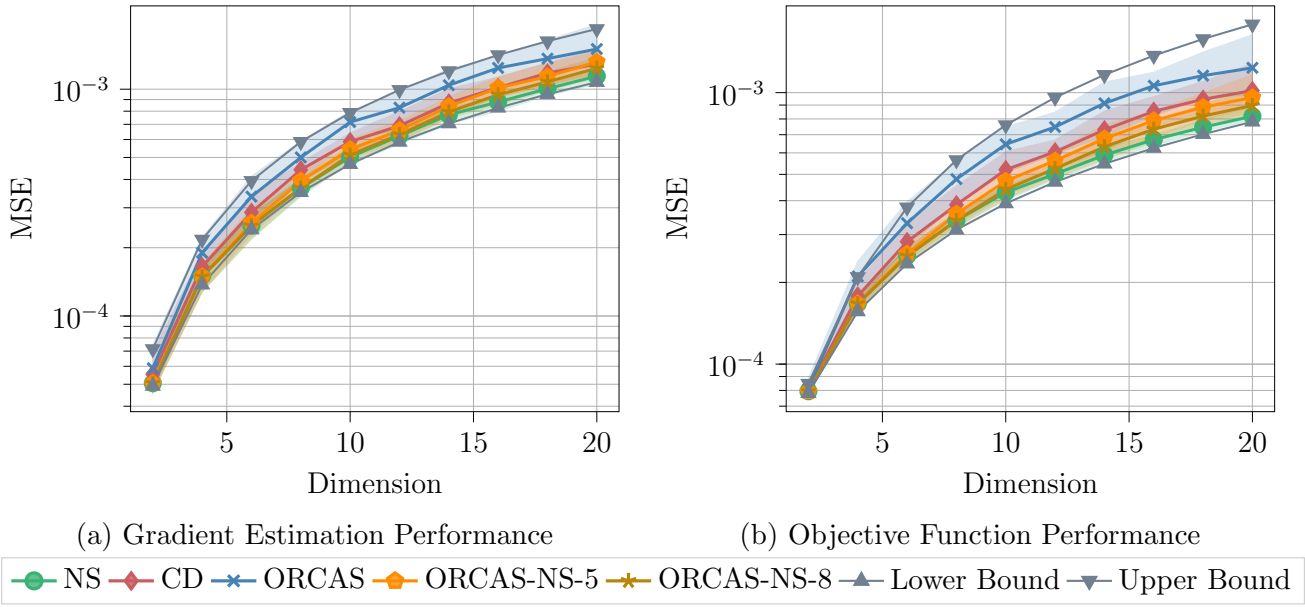


(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.10: Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with an increasing condition number and in the performance regime $\tilde{h} \ll h$.

Performance Regime: $\tilde{h} \approx h$

In Figure 4.11 we see the performance gap reducing drastically again. As in the fixed condition number case, `ORCAS` performs slightly worse than the other methods due to it aligning its final sample point more closely along the sensitive direction.

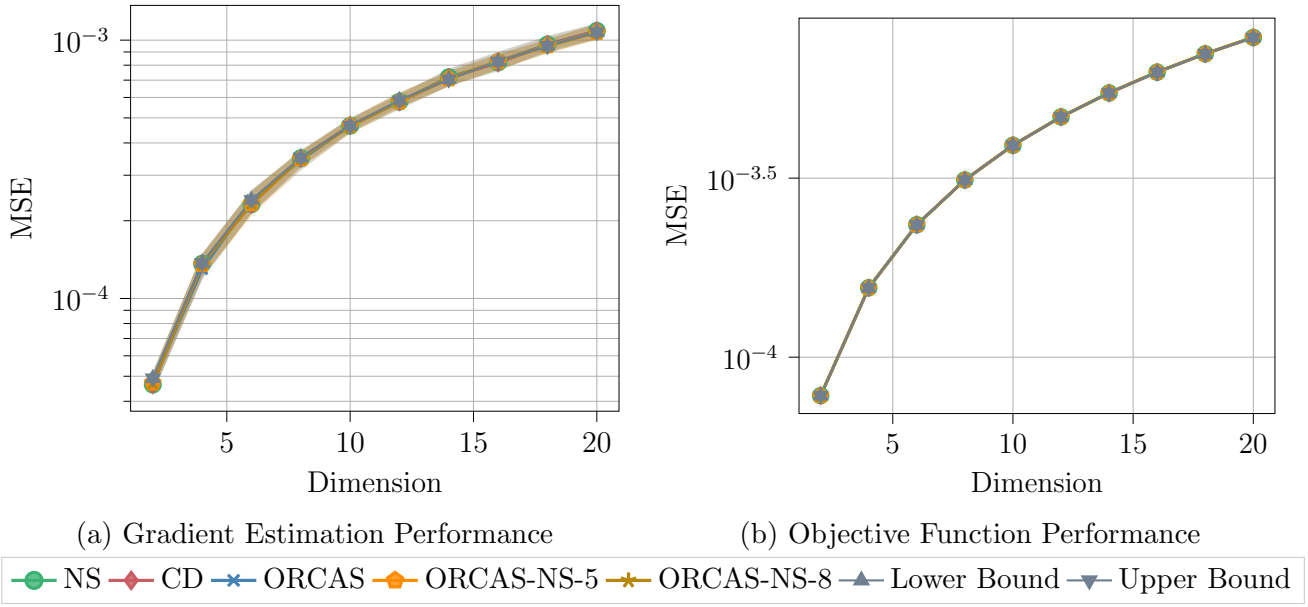


(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.11: Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with an increasing condition number and in the performance regime $\tilde{h} \approx h$.

Performance Regime: $\tilde{h} \gg h$

As before, we see in Figure 4.12 no performance difference between the methods. This is because the higher-order errors are dominating and hence reducing the lower-order errors has little effect.



(a) Gradient Estimation Performance (b) Objective Function Performance

Figure 4.12: Mean squared error as a function of dimension for the performance for ill-conditioned polynomials with an increasing condition number and in the performance regime $\tilde{h} \gg h$.

4.7 Numerics: Estimated Third Derivatives via a Global Model

In the previous Section, we only focused on the case when the third derivative is available exactly. This is not a realistic situation, as it is unlikely that gradient estimation is unavailable in the presence of third order information. To assess the performance of the above methods, we consider a framework that uses a surrogate model built from historical function evaluations to obtain approximate third order information. Such a set-up is sensible as in many real world situations some historical function evaluations are already available and can then be used for further local computations and explorations. We then view this Section as a proof of concept for the viability of such an approach.

The core of this framework rests on the idea that global information, given by historical function

evaluations, can inform the selection of local sample points. A local method will generally always obtain more information for a gradient estimate than global information alone, however, without any knowledge of the function itself the sample point selection may be very poor. Given this perspective, it becomes more clear why combining both methods may be beneficial over using each individually. We will confirm this by comparing methods using a global model to the gradient estimates given directly by the global model and to a baseline estimator which does not use any global model information.

The key difficulty is that it is not clear how accurate the information obtained by the global model is. Methods such as the numerical solver may overfit to the structure of the inaccurate third derivative. More concretely, the sample set may sacrifice orthogonality, and hence incur a higher noise error, in order to achieve a lower approximation error. However, as the approximation error information given by the third derivative is itself inaccurate, this may be done in vain. Therefore, more robust structures, such as ORCAS, which are biased towards orthogonality may be useful. We note that this observation also justifies the use of a linear model for gradient estimation, as they are more robust to overfitting.

The final observation relates to the use of the hyperparameter $h^{(HP)}$ used in the objective function given in Definition 54. We have had extensive discussions on the interpretation of this parameter as approximating higher order errors as seen in the case study in Section 4.2.4. However, when using inaccurate third order derivatives, it can also be interpreted as a tuning parameter on the radius of the sample set in accordance with the third order error. This is because an estimated third order derivative may provide an inaccurate estimate of the size of the approximation error. When the size is overestimated the effect is clear, the bound simply enforces a solution to have the correct smaller radius. In the case when the size is underestimated the bound may still be useful. Specifically, methods which are designed to find roots will scale each difference vector to have a size given by the hyperparameter. Then, even if the sample point is not a root for the true third order derivative, the corresponding difference vector will have the correct size as given by the hyperparameter.

We will not estimate the noise level in these experiments. This choice was made to focus on the

more difficult problem of estimating the higher order information. A strategy to estimate the noise level may be a Monte Carlo estimate at a given location. However, the specific method will inherently depend on the application use-case.

4.7.1 Global Model Framework

We use the global model framework from [LPKJ23]. In summary, what we do is sample some number of points from a set domain uniformly. We then fit a global model to it. We choose use a cubic-spline model as it provides a good trade-off between flexibility and robustness. We then simply use the third order derivative from this model as an estimate of the true third order derivative. We use the implementation by `scipy` [VGO+20] and extend it to allow for the analytic computation of the gradient and third order information once the model has been fit. We do not using a smoothing parameter since all points are sufficiently far apart.

Once a global model has been fit, we use it for a number of gradient estimates on a fixed domain of the function. While possible, we will not refit the global model given any new information incurred during gradient estimation to avoid any path-dependencies between the models. Other use-cases, such as in optimization, may benefit from different strategies. However, here we want to focus solely on the performance of each method as a gradient estimator.

4.7.2 Numerical Results: Ackley

The Ackley function is a common test function due to its high degree of non-linearity and flexibility in scaling to higher dimensions. It is given by

$$f(x_0 \cdots x_n) = -20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)) + 20 + e.$$

Notably, it is not-differentiable at 0, so the function will not be evaluated there.

Set-Up

We use the following setup for the experiments.

- We consider dimensions given by $d \in \{4, 8, 12\}$.
- The domain is given by the unit cube $[-0.5, 0.5]^{\otimes d}$.
- The number of global model fitting points is $N^{(GM)} \in \{100, 250, 500, 750, 1000, 1250, 1500\}$. These are sampled at the beginning of the experiment and the global model is fixed for the remainder. The fitting points are sampled uniformly on the entire domain.
- One hundred evaluation points are used; that is the points at which each gradient estimation method is evaluated. They are sampled uniformly on $[-0.45, 0.45]^{\otimes d}$. This is to mitigate any boundary effects arising from the global model.
- We let $h^{(HP)} \in \{0.1, 0.075, 0.05, 0.025\}$. At every evaluation point we retain the best performing hyperparameter.
- The noise level is set to 0.001 as in the previous section.

Gradient Estimation Methods

We use all the gradient estimation methods that we have already introduced in Section 4.6.2 except for the lower and upper bounds as we do not have access to their analytic computation. Instead, we introduce a `Basline` method, which uses the sample set $h^{(HP)} I_{d \times d}$. This baseline assesses the performance of a method that does not use any global information. Generally this would be called central differences, however, we reserve the name to the method which adjusts the difference vectors according to higher order information.

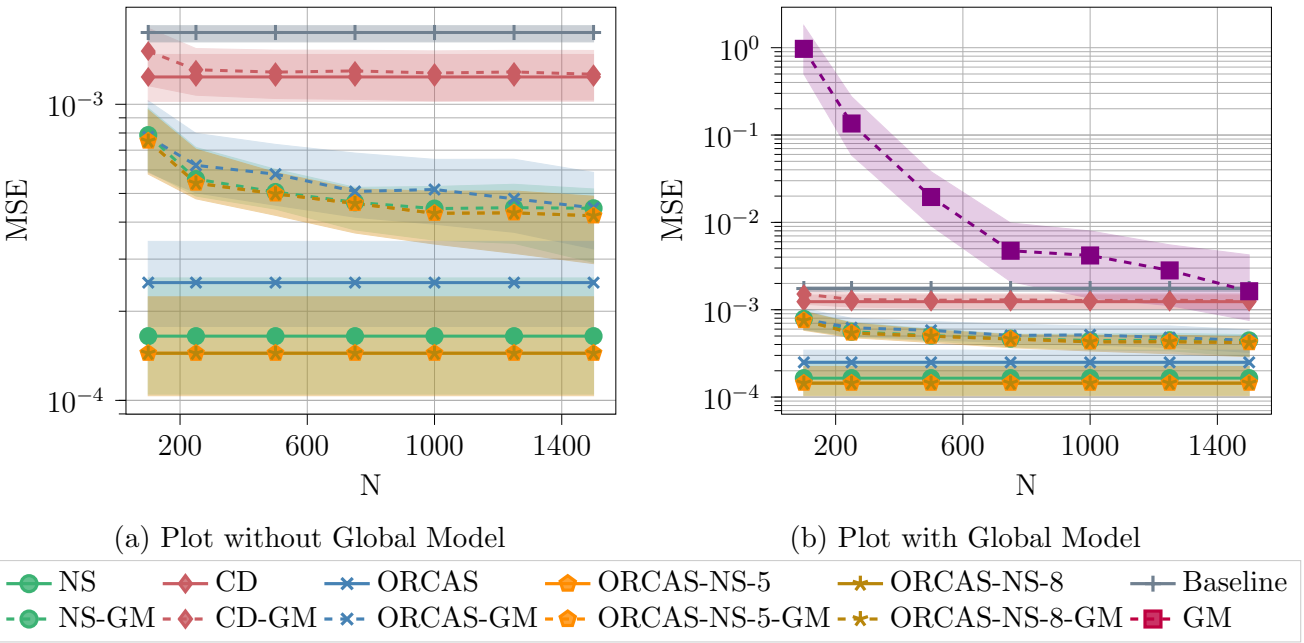
Any method that uses a global model estimate has a `-GM` appended. The method without this suffix use the exact third order derivatives to provide a best case comparison. With these methods we can assess how much better a method could be if the third order information was more accurate.

Lastly, we denote the gradient estimate that arises directly from the global model simply by `GM`. Note, this method does not receive any further local samples as there is no well-defined sample set construction strategy for this case.

Observations: Ackley in Dimension 4

In Figure 4.13 we see that almost all methods using exact higher-order information are roughly an order of magnitude better than the baseline. The exception is `CD` which barely improves on the baseline. As the global model improves with increasing number of samples points, the models using the global model also improve significantly. They appear to approach the performance of the methods using the exact higher-order information before plateauing. The difference in performance between `ORCAS` and numerical solvers is marginal when using the global model. The benefit of slightly better performance is removed by the inaccuracy due to the higher-order estimates.

Notably, the global model improves quickly. Towards $1,500$ fitting points it also starts to improve on the `Baseline`.



(a) Plot without Global Model

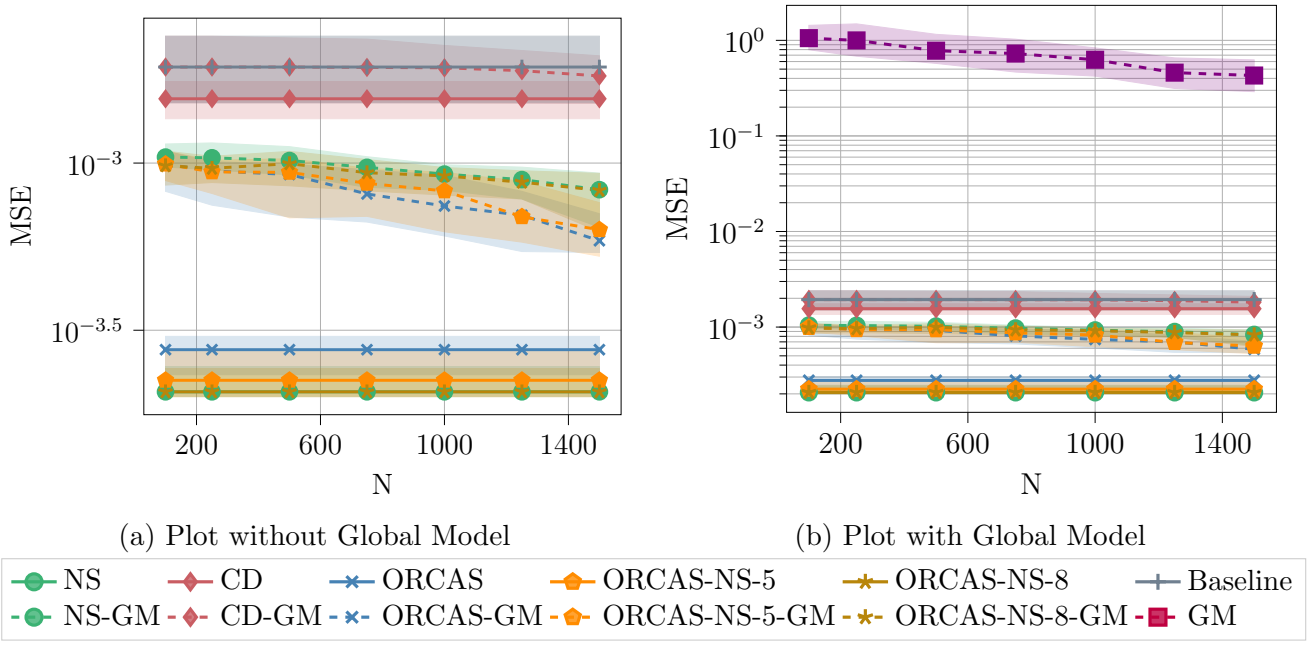
(b) Plot with Global Model

Figure 4.13: Mean squared error as a function of the number of sample points used to fit the global model for the Ackley function in dimension $d = 4$.

Observations: Ackley in Dimension 8

The observations for Figure 4.14 are similar to the case when the dimension is four but due to the higher dimensionality the global model is unable to improve sufficiently. Nevertheless, some improvement can be observed for all models with an increasing number of sample points. Importantly, even with very few sample points, methods using the global model outperform the baseline.

Interestingly, `ORCAS` is now doing better than the numerical solver when using global model information. Here we may see the first signs of the numerical solver overfitting to a noisy estimate.



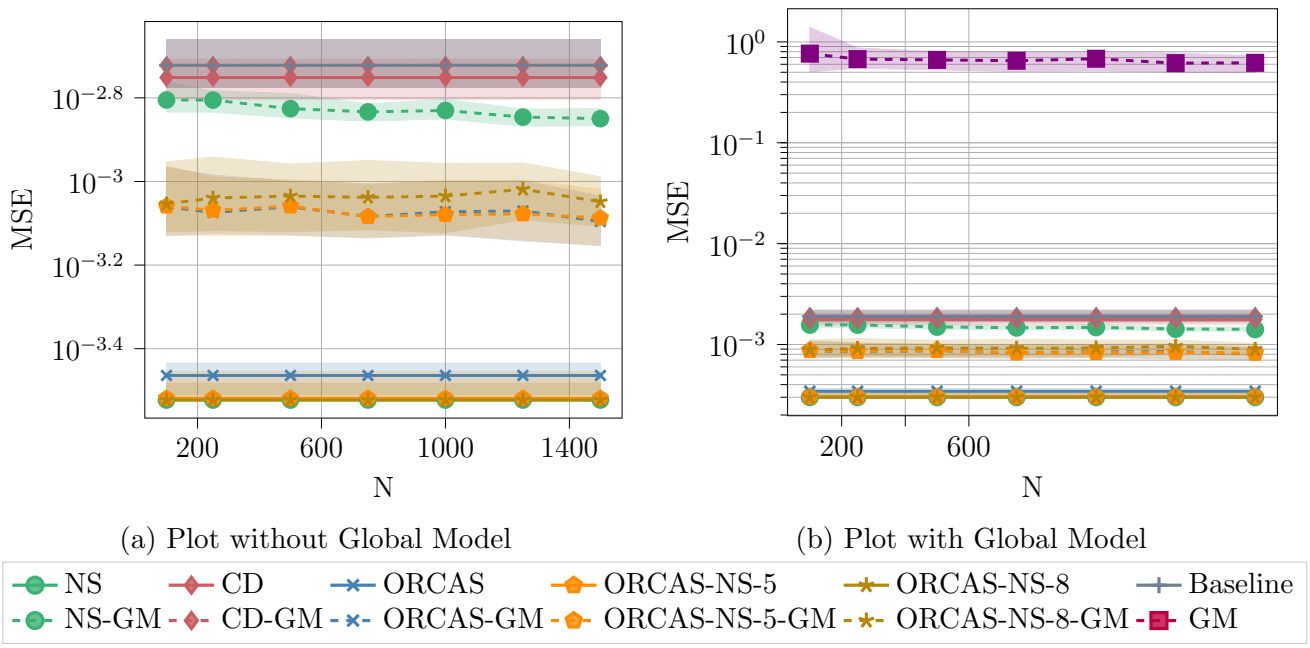
(a) Plot without Global Model

(b) Plot with Global Model

Figure 4.14: Mean squared error as a function of the number of sample points used to fit the global model for the Ackley function in dimension $d = 8$.

Observations: Ackley in Dimension 12

In Figure 4.15 we see essentially no improvement in the global model with more datapoints. Still, the models using the global model remain better than the baseline. Crucially, `ORCAS`-based models using the global model are now distinctly better than the numerical solver, further confirming that the numerical solver is overfitting.



(a) Plot without Global Model

(b) Plot with Global Model

Figure 4.15: Mean squared error as a function of the number of sample points used to fit the global model for the Ackley function in dimension $d = 12$.

4.8 Proofs

Proof. We expand the means squared error after which we proceed term by term

$$\begin{aligned}\mathbb{E}[\|\nabla_S \tilde{f}(x') - \nabla f(x')\|_2^2] &= \mathbb{E}[\|\nabla_S \tilde{f}(x') - \nabla_S f(x') + \nabla_S f(x') - \nabla f(x')\|_2^2] \\ &= \mathbb{E}[\|\nabla_S \tilde{f}(x') - \nabla_S f(x')\|_2^2] + \|\nabla_S f(x') - \nabla f(x')\|_2^2 \\ &\quad + 2\mathbb{E}[\Big(\nabla_S \tilde{f}(x') - \nabla_S f(x')\Big)^T \Big(\nabla_S f(x') - \nabla f(x')\Big)].\end{aligned}$$

By linearity of the simplex gradient we have that $\mathbb{E}[\|\nabla_S \tilde{f}(x') - \nabla_S f(x')\|_2^2] = \mathbb{E}[\|\nabla_S \epsilon(0)\|_2^2]$

where we write $\nabla_S \epsilon(0) = (S^T)^+ \epsilon(\mathcal{X})$ with $\epsilon(\mathcal{X}) = [\epsilon(1), \ldots, \epsilon(2d)]^T$ [4]. We then have

$$\begin{aligned}\mathbb{E}[\|\nabla_S \epsilon(0)\|_2^2] &= \mathbb{E}[\epsilon(\mathcal{X})^T S^+ (S^T)^+ \epsilon(\mathcal{X})] \\ &= \operatorname{tr}\left(S^+ (S^T)^+ \sigma^2\right) \\ &= \sigma^2 \|S^+\|_F^2\end{aligned}$$

where we used the result from [Ken05] that for any random vector $\mathbf{X}$, with zero mean and covariance matrix Σ, and symmetric matrix A we have $\mathbb{E}[\mathbf{X}^T A \mathbf{X}] = \operatorname{tr}(A\Sigma)$.

Lastly, the second term $\|\nabla_S f(x') - \nabla f(x')\|_2^2$ does not need to be further reduced. And, the cross term $\mathbb{E}[\left(\nabla_S \tilde{f}(x') - \nabla_S f(x')\right)^T \left(\nabla_S f(x') - \nabla f(x')\right)]$ is easily seen to equal zero since $\mathbb{E}[\nabla_S \tilde{f}(x')] = \nabla_S f(x')$. □

4.8.1 Extended Case Study Results

Lemma 83. *Let $M_{\hat{S}}^{(3,5)} = \zeta_{\hat{S}} M_{\hat{S}}^{(3)} M_{\hat{S}}^{(5)}$ and $r = \eta_r h_{\hat{S}}^*$. If $h_{\hat{S}} = h_{\hat{S}}^*$, and $h_{\hat{S}} = \gamma \tilde{h}_{\hat{S}}$ then we have*

$$\ell_{p,\sigma}^{(GE)}(r, \hat{S}) = \frac{\tau_{\hat{S}}}{(\eta_r h)^2}\left(\frac{(\gamma\eta_r)^6}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3\eta_r^8 + \frac{1}{4}\eta_r^{10} + 1\right).$$

Otherwise, when $\tilde{h}_{\hat{S}} = h_{\hat{S}}^*$ and $\tilde{h}_{\hat{S}} = \gamma h_{\hat{S}}$ then we have

$$\ell_{p,\sigma}^{(GE)}(r, \hat{S}) = \frac{\tau_{\hat{S}}}{(\eta_r \tilde{h}_{\hat{S}})^2}\left(\frac{\eta_r^6}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5\eta_r^8 + \frac{1}{4}\gamma^{10}\eta_r^{10} + 1\right).$$

Lemma 83. For the first part we have

$$\begin{aligned}\ell_{p,\sigma}^{(GE)}(r, \hat{S}) &= \frac{1}{r^2}\left(r^6 (M_{\hat{S}}^{(3)})^2 + 2r^8 M_{\hat{S}}^{(3,5)} + r^{10} M_{\hat{S}}^{(5)} + \tau_{\hat{S}}\right) \\ &= \frac{\tau_{\hat{S}}}{(\eta_r h)^2}\left(\frac{(\gamma\eta_r)^6}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3\eta_r^8 + \frac{1}{4}\eta_r^{10} + 1\right).\end{aligned}$$

[4] We abuse notation slightly, and see ϵ as both a function of time with domain $\mathbb{R}$ and input domain $\mathbb{R}^d$. The idea is that if we evaluate the noise at different locations, in practice there is a temporal aspect to this, as the function cannot be evaluated at both locations at the same time.

For the second part we have

$$\begin{aligned}\ell_{p,\sigma}^{(GE)}(r,\hat{S}) &= \frac{1}{r^2}\left(r^6(M_{\hat{S}}^{(3)})^2 + 2r^8 M_{\hat{S}}^{(3,5)} + r^{10}M_{\hat{S}}^{(5)} + \tau_{\hat{S}}\right)\\ &= \frac{\tau_{\hat{S}}}{(\eta_r\tilde{h}_{\hat{S}})^2}\left(\frac{\eta_r^6}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5\eta_r^8 + \frac{1}{4}\gamma^{10}\eta_r^{10} + 1\right).\end{aligned}$$

□

Lemma 84 (Extended Lemma 57)**.** *Let a $\ell_{p,\sigma}^{(GE)}$ and $\hat{S}$ be given. If $h_{\hat{S}} \geq \tilde{h}_{\hat{S}}$ and letting $\gamma\tilde{h}_{\hat{S}} = h_{\hat{S}}$ we have that $\frac{|\tilde{h}_{\hat{S}} - r_{\hat{S}}^*|}{\tilde{h}_{\hat{S}}} = \mathcal{O}(\zeta_{\hat{S}}\gamma^5) + \mathcal{O}(\gamma^{10})$ with a tighter bound given by*

$$\tilde{h}_{\hat{S}}\max(0.83, (\frac{1}{1+\zeta_{\hat{S}}\frac{3}{\sqrt{2}}\gamma^5 + \gamma^{10}})^{1/6}) \leq r_{\hat{S}}^* \leq \tilde{h}_{\hat{S}}.$$

When $h_{\hat{S}} \leq \tilde{h}_{\hat{S}}$ and $\gamma h_{\hat{S}} = \tilde{h}_{\hat{S}}$ we have that $\frac{|h_{\hat{S}} - r_{\hat{S}}^*|}{h_{\hat{S}}} = \mathcal{O}(\zeta_{\hat{S}}\gamma^3) + \mathcal{O}(\gamma^6)$ with a tighter bound given by

$$h_{\hat{S}}\max(0.83, \left(1 - \frac{1}{2}(\gamma^6 + \zeta_{\hat{S}}\frac{3}{\sqrt{2}}\gamma^3)\right)^{1/5}) \leq r_{\hat{S}}^* \leq h_{\hat{S}}.$$

Lemma 84. The derivative of Equation 4.2.1 with respect to r for a given $\hat{S}$ is given by

$$\frac{d\ell_{p,\sigma}^{(GE)}(r,\hat{S})}{dr} = (M_{\hat{S}}^{(3)})^2 r^3 + 12\zeta_{\hat{S}}M_{\hat{S}}^{(3)}M_{\hat{S}}^{(5)}r^5 + 8(M_{\hat{S}}^{(5)})^2 r^7 - 2\frac{\tau_{\hat{S}}}{r^3}. \tag{4.8.1}$$

First we consider $h_{\hat{S}} \geq \tilde{h}_{\hat{S}}$ which gives us $\gamma h_{\hat{S}} = \tilde{h}_{\hat{S}}$. We want to find η such that $r_{\hat{S}}^* = \eta\tilde{h}_{\hat{S}}$ for $0 \leq \gamma \leq 1$. Plugging $\eta\tilde{h}_{\hat{S}}$ into Equation 4.8.1 and setting to zero we get

$$\eta^6 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\eta^8\gamma^5 + \eta^{10}\gamma^{10} - 1 = 0.$$

If we let $\eta = 1$ then the left hand side will be greater than one. Now, consider

$$\eta^6 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\eta^8\gamma^5 + \eta^{10}\gamma^{10} - 1 \leq \eta^6\left(1 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5 + \gamma^{10}\right) - 1$$

and let $\eta^{(lb)} = \left(1 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5 + \gamma^{10}\right)^{-1/6}$ making the left hand-side equal to zero. Hence, we know

that $\eta^{(lb)}\tilde{h}_{\hat{S}} \leq r^*_{\hat{S}} \leq \tilde{h}_{\hat{S}}$ by the intermediate value-theorem. Using a first-order Taylor-bound we have that $\eta^{(lb)} \geq 1 - \frac{1}{6}\left(\frac{3}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5 + \gamma^{10})\right)$ which provides the order terms.

We proceed similarly for the case $h_{\hat{S}} \leq \tilde{h}_{\hat{S}}$ where we say $h = \gamma\tilde{h}$. Plugging in $\eta h_{\hat{S}}$ into the derivative and setting to zero we have

$$\eta^6\gamma^6 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\eta^8\gamma^3 + \eta^{10} - 1 = 0.$$

Letting $\eta = 1$ it is again clear that the left hand side is greater than zero. Now consider

$$\eta^6\gamma^6 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\eta^8\gamma^3 + \eta^{10} - 1 \geq \eta^5\left(\gamma^6 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3\right) + \eta^{10} - 1$$

where we can solve the right-hand side to equal zero. We do so by substituting η^5 and using the quadratic formula to obtain $\left((-(\gamma^6 + \zeta_{\hat{S}}\frac{3}{\sqrt{2}}\gamma^3) + \sqrt{(\gamma^6 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3)^2 + 4})\frac{1}{2}\right)^{1/5}$ which we lower-bound with $\left(1 - \frac{1}{2}(\gamma^6 + \frac{3}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3)\right)^{1/5} = \eta^{(lb)}$. We then have $\eta^{(lb)}h_{\hat{S}} \leq r^*_{\hat{S}} \leq h_{\hat{S}}$. Using a first-order Taylor bound we have that $|1 - \eta^{(lb)}| = \mathcal{O}(\zeta_{\hat{S}}\gamma^3) + \mathcal{O}(\gamma^6)$.

To see that $0.83\ h^*_{\hat{S}} \leq r^*_{\hat{S}}$ for all γ and $\zeta_{\hat{S}}$ note that setting $\gamma = 1$ we upper-bound the derivative for all η. Setting $\eta = 0.83$ we obtain an upper-bound that is less than zero, and since the function is increasing this implies that the true root must have been bigger. □

Lemma 85 (Restated Lemma 58)**.** *Let $\ell^{(GE)}_{p,\sigma}$ and $\hat{S}$ be given. Then we have that*

$$\left|\frac{\ell^{(GE)}_{p,\sigma}(\tilde{h}_{\hat{S}}, \hat{S})}{\ell^{(GE)}_{p,\sigma}(r^*_{\hat{S}}, \hat{S})} - 1\right| = \mathcal{O}\left(\left(\frac{|h^*_{\hat{S}} - r^*_{\hat{S}}|}{h^*_{\hat{S}}}\right)^2\right).$$

Lemma 58. For the first part let $r^*_{\hat{S}} = \eta\tilde{h}_{\hat{S}}$. Using Lemma 83 we can write

$$\frac{\ell^{(GE)}_{p,\sigma}(\tilde{h}_{\hat{S}}, \hat{S})}{\ell^{(GE)}_{p,\sigma}(r^*_{\hat{S}}, \hat{S})} = \eta^2\frac{(\frac{1}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5 + \gamma^{10} + 1)}{(\frac{\eta^6_{r^*_{\hat{S}}}}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5\eta^8 + \frac{1}{4}\gamma^{10}\eta^{10} + 1)}.$$

We now bring everything under the same denominator and focus on the numerator. For this, let $\eta = 1 - \epsilon_\eta$ justified by Lemma 57 given that $\eta \leq 1$. We expand all powers of η to first order,

where the higher order terms are collected in R_η, yielding

$$\begin{aligned}
&\eta^2(\frac{1}{2}+\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5+\frac{1}{4}\gamma^{10}+1)-(\frac{\eta^6}{2}+\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5\eta^8+\frac{1}{4}\gamma^{10}\eta^{10}+1)\\
&=(1-2\epsilon_\eta)(\frac{1}{2}+\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5+\frac{1}{4}\gamma^{10}+1)-(1-6\epsilon_\eta)\frac{1}{2}-(1-8\epsilon_\eta)\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5-(1-10\epsilon_\eta)\frac{1}{4}\gamma^{10}-1+R_\eta\\
&=\frac{6}{\sqrt{2}}\epsilon_\eta\zeta_{\hat{S}}\gamma^5+2\epsilon_\eta\gamma^{10}+R_\eta\\
&=\mathcal{O}(\epsilon_\eta\zeta_{\hat{S}}\gamma^5)+\mathcal{O}(\epsilon_\eta\gamma^{10})\\
&=\mathcal{O}(\zeta_{\hat{S}}^2\gamma^{10})+\mathcal{O}(\zeta_{\hat{S}}\gamma^{15})+\mathcal{O}(\gamma^{20})=\mathcal{O}(\epsilon_\eta^2)
\end{aligned}$$

where the last lines follow from Lemma 57. The denominator can be bounded by a constant, which yields the first result.

We proceed similarly for the second result, where we assume that $r^*_{\hat{S}}=\eta h_{\hat{S}}$ providing

$$\frac{\ell^{(GE)}_{p,\sigma}(h_{\hat{S}},\hat{S})}{\ell^{(GE)}_{p,\sigma}(r^*_{\hat{S}},\hat{S})}=\eta^2\frac{(\frac{1}{2}\gamma^6+\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3+\frac{1}{4}+1)}{(\frac{\eta^6}{2}\gamma^6+\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3\eta^8+\frac{1}{4}\eta^{10}+1)}.$$

We proceed as before letting $\eta=1-\epsilon_\eta$

$$\begin{aligned}
&\eta^2(\frac{1}{2}\gamma^6+\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3+\frac{1}{4}+1)-(\frac{\eta^6}{2}\gamma^6+\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3\eta^8+\frac{1}{4}\eta^{10}+1)\\
&=(1-2\epsilon_\eta)(\frac{1}{2}\gamma^6+\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5+\frac{1}{4}+1)-(1-6\epsilon_\eta)\frac{1}{2}\gamma^6-(1-8\epsilon_\eta)\frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5-(1-10\epsilon_\eta)\frac{1}{4}-1+R_\eta\\
&=4\epsilon_\eta\gamma^6+\frac{6}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3+R_\eta\\
&=\mathcal{O}(\epsilon_\eta\gamma^6)+\mathcal{O}(\epsilon_\eta\zeta_{\hat{S}}\gamma^3)\\
&=\mathcal{O}(\gamma^{12})+\mathcal{O}(\zeta_{\hat{S}}\gamma^9)+\mathcal{O}(\zeta_{\hat{S}}^2\gamma^6)=\mathcal{O}(\epsilon_\eta^2)
\end{aligned}$$

where the last line again follow from Lemma 57. Bounding the denominator yields the result. □

Lemma 86 (Extended Lemma 60)**.** *Let $\underline{\hat{S}}$ be a baseline sample set and $\hat{S}^*$ be the optimal sample set as described above. Assume that $\tilde{h}\leq h$ and $\tilde{h}=\gamma h$ then*

$$\frac{\ell^{(GE)}_{p,\sigma}(\tilde{h},\underline{\hat{S}})}{\ell^{(GE)}_{p,\sigma}(h,\hat{S}^*)}\leq\frac{1}{\gamma^2}\frac{1}{5}(6+2\sqrt{2}\zeta_{\hat{S}}\gamma^5+\gamma^{10}).$$

Assume $\tilde{h} \geq h$ *and* $h_{\hat{S}} = \gamma \tilde{h}_{\hat{S}}$ *then we have*

$$\left| \frac{\ell_{p,\sigma}^{(GE)}(h, \hat{S})}{\ell_{p,\sigma}^{(GE)}(h, \hat{S}^*)} - 1 \right| \leq \frac{2}{5}\gamma^6 + \frac{2\sqrt{2}}{5}\zeta_{\hat{S}}\gamma^3.$$

Lemma 86. The result follows from a simple application of Lemma 83. □

Lemma 87 (Restated Lemma 61)**.** *When* $h \geq \tilde{h}_{\hat{S}}$ *we have that* $\frac{\ell_{p,\sigma}^{(GE)}(\min(h,\tilde{h}_{\hat{S}}),\hat{S})}{\ell_{p,\sigma}^{(GE)}(r^*_{\hat{S}},\hat{S})} = 1$*. When* $h \leq \tilde{h}_{\hat{S}} \leq h_{\hat{S}}$ *we have* $\frac{\ell_{p,\sigma}^{(GE)}(h,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h^*_{\hat{S}},\hat{S})} = \mathcal{O}(\gamma^2\kappa^{1/5})$*. When* $h \leq h_{\hat{S}} \leq \tilde{h}_{\hat{S}}$ *we have* $\frac{\ell_{p,\sigma}^{(GE)}(h,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h^*_{\hat{S}},\hat{S})} = \mathcal{O}(\kappa^{1/5})$*. Lastly, when* $h_{\hat{S}} \leq h \leq \tilde{h}_{\hat{S}}$ *we have* $\frac{\ell_{p,\sigma}^{(GE)}(h,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h^*_{\hat{S}},\hat{S})} = \mathcal{O}(\kappa^{4/5})$*.*

Lemma 61. The proof will proceed by cases, where we have six different combinations in which to order h, $h_{\hat{S}}$ and $\tilde{h}_{\hat{S}}$.

Consider the cases when $h \geq \tilde{h}_{\hat{S}}$ and $h_{\hat{S}} \geq \tilde{h}_{\hat{S}}$ then trivially we have $\frac{\ell_{p,\sigma}^{(GE)}(\min(h,\tilde{h}_{\hat{S}}),\hat{S})}{\ell_{p,\sigma}^{(GE)}(h^*_{\hat{S}},\hat{S})} = 1$.

For the case when $h_{\hat{S}} \leq \tilde{h}_{\hat{S}} \leq h$ we have $\frac{\ell_{p,\sigma}^{(GE)}(\tilde{h}_{\hat{S}},\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}},\hat{S})}$ where $\frac{h_{\hat{S}}}{\gamma} = \tilde{h}_{\hat{S}}$. We now expand using Lemma 83

$$\frac{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}/\gamma,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}},\hat{S})} = \gamma^2 \frac{\frac{1}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^{-5} + \gamma^{-10}\frac{1}{4} + 1}{\gamma^6\frac{1}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3 + \frac{1}{4} + 1}$$

where bounding the denominator we have that $\frac{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}/\gamma,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}},\hat{S})} = \mathcal{O}(\frac{1}{\gamma^8})$. Given that $\tilde{h}_{\hat{S}} \leq h$ and $h\frac{1}{\kappa^{1/10}} \geq h_{\hat{S}}$ we have that $\tilde{h}_{\hat{S}} = \frac{h_{\hat{S}}}{\gamma} \leq h \leq \kappa^{1/10}h_{\hat{S}}$ yielding $\frac{1}{\gamma} \leq \kappa^{1/10}$ and hence $\frac{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}/\gamma,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}},\hat{S})} = \mathcal{O}(\kappa^{4/5})$.

Now, consider $h_{\hat{S}} \leq h \leq \tilde{h}_{\hat{S}}$ which follows almost the identical analysis to the above only replacing $\frac{1}{\gamma}$ with $\kappa^{1/10}$ and hence we have $\frac{\ell_{p,\sigma}^{(GE)}(h,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}},\hat{S})} = \mathcal{O}(\kappa^{4/5})$.

Next, consider $h \leq h_{\hat{S}} \leq \tilde{h}_{\hat{S}}$. We have that $h\kappa^{1/10} \leq h_{\hat{S}}$. Since the objective function is convex in r we have that $\ell_{p,\sigma}^{(GE)}(h,\hat{S}) \leq \ell_{p,\sigma}^{(GE)}(h_{\hat{S}},\hat{S})$. Then we have

$$\frac{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}}/\kappa^{1/10},\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}},\hat{S})} = \kappa^{1/5}\frac{\gamma^6\kappa^{-6/10}\frac{1}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3\kappa^{-8/10} + \kappa^{-1}\frac{1}{4} + 1}{\gamma^6\frac{1}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^3 + \frac{1}{4} + 1}$$

which yields $\frac{\ell_{p,\sigma}^{(GE)}(h,\hat{S})}{\ell_{p,\sigma}^{(GE)}(h_{\hat{S}},\hat{S})} = \mathcal{O}(\kappa^{1/5})$.

Lastly, consider $h \leq \tilde{h}_{\hat{S}} \leq h_{\hat{S}}$. Here we have that $h \leq \frac{1}{\kappa^{1/10}} h_{\hat{S}} = \frac{1}{\kappa^{1/10}} \frac{1}{\gamma} \tilde{h}_{\hat{S}}$. Then we have that

$$\frac{\ell_{p,\sigma}^{(GE)}(h, \hat{S})}{\ell_{p,\sigma}^{(GE)}(\tilde{h}_{\hat{S}}, \hat{S})} \leq \gamma^2 \kappa^{1/5} \frac{\gamma^{-6}\kappa^{-6/10}\frac{1}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^{-3}\kappa^{-8/10} + \kappa^{-1}\frac{1}{4} + 1}{\frac{1}{2} + \frac{1}{\sqrt{2}}\zeta_{\hat{S}}\gamma^5 + \frac{1}{4}\gamma^{10} + 1}.$$

We can again bound the denominator. Also, notice that $0 \leq \gamma^{-6}\kappa^{-6/10}, \gamma^{-3}\kappa^{-8/10} \leq 1$ since $\gamma \leq \frac{1}{\kappa^{1/10}}$ due to $h \leq \tilde{h}_{\hat{S}}$. Hence, we have that $\frac{\ell_{p,\sigma}^{(GE)}(h, \hat{S})}{\ell_{p,\sigma}^{(GE)}(\tilde{h}_{\hat{S}}, \hat{S})} = \mathcal{O}(\gamma^2 \kappa^{1/5})$. □

4.8.2 Variance of ill-conditioned polynomial

To prove the following results we will be working in hyper-spherical coordinates. For this we will introduce some notation and preliminary results.

We have a set of variables r and $\phi_1, \ldots, \phi_{d-1}$. The mapping is then given by $x_1 = r\cos\phi_1$, $x_2 = r\sin\phi_1\cos\phi_2$, $\cdots$, $x_d = r\sin\phi_1 \cdots \sin\phi_{d-2}\sin\phi_{d-1}$. The volume element is given by

$$dV = r^{d-1}\sin^{d-2}\phi_1 \cdots \sin\phi_{d-2} dr d\phi_1 \cdots d\phi_{d-1}.$$

We denote this surface area by S_{d-1}. The closed-form solution to this is $S_{d-1} = \frac{2\pi^{d/2}}{\Gamma(\frac{d}{2})}$ with Γ being the gamma function. The only property we need for the Γ-function is that $\Gamma(z+1) = z\Gamma(z)$

Notice that

$$\int_0^{2\pi} \cdots \int_0^{\pi} \sin^{d-2}\phi_1 \cdots \sin\phi_{d-2} d\phi_1 \cdots d\phi_{d-1}$$

is simply the surface area of a $(d-1)$-ball.

Lemma 88. *The following relations hold for the Γ-function and the surface area of a ball*

- $\frac{\Gamma(i+j)}{\Gamma(i)} = (i)(i+1)\cdots(i+j-1)$,
- $\frac{S_{(i+2j)}}{S_{i-1}} = (2\pi)^j \frac{1}{i(i+2)\cdots(i+2(j-1))}$.

Proof. The first result follows simply by applying the base case given by applying $\Gamma(z+1) = z\Gamma(z)$ and applying the appropriate induction hypothesis

$$\begin{aligned}\frac{\Gamma(i+j)}{\Gamma(i)} &= \frac{\Gamma(i+j)}{\Gamma(i+j-1)}\frac{\Gamma(i+j-1)}{\Gamma(i)} \\ &= (i+j-1)\frac{\Gamma(i+j-1)}{\Gamma(i)}.\end{aligned}$$

We now use the above result to prove the second statement. Here we have

$$\begin{aligned}\frac{S_{(i+2j)-1}}{S_{i-1}} &= \frac{2\pi^{\frac{(i+2j)}{2}}}{\Gamma(\frac{(i+2j)}{2})}\frac{\Gamma(\frac{i}{2})}{2\pi^{\frac{i}{2}}} \\ &= \pi^j\frac{\Gamma(\frac{i}{2})}{\Gamma(\frac{(i+2j)}{2})} \\ &= \pi^j\frac{2}{i}\frac{2}{i+2}\cdots\frac{2}{i+2(j-1)}.\end{aligned}$$

□

Lemma 89 (Restated Lemma 79)**.** *Let $X \sim \mathcal{N}(0, \frac{1}{\sqrt{d}}I_{d\times d})$ and p be a random polynomial of order k such that $\mathbb{V}_{X,P}[P(X)] = M^2$ and $P(X) \sim P(x')$ if $\|x\| = \|x'\|$. Then for any unit vector x we have*

$$\mathbb{V}_P[P(X)] = M^2\frac{d^k}{d(d+2)\cdots(d+2(k-1))}.$$

Lemma 79. Writing $X = \|X\|\hat{X}$ and by the law of total variance we have that $\mathbb{V}_{X,P}[P(X)] = \mathbb{E}_X[\|X\|^{2k}\mathbb{V}_P[P(\hat{X})]$. Since $P(X) \sim P(x')$ if $\|x\| = \|x'\|$ and since the random vector $\hat{X}$ is a unit-vector we have that $\mathbb{V}_P[P(\hat{X})]$ is a constant which we denote by Ψ. Now, computing this term, Ψ, will provide the result

We write the expression $\Psi\mathbb{E}_X[\|X\|^{2k}] = M^2$ in hyper-spherical coordinates to obtain

$$\Psi\left(\int_0^\infty \frac{1}{(2\pi\sigma^2)^{d/2}}r^{2k}e^{-\frac{r^2}{2\sigma^2}}r^{d-1}dr\right)S_{d-1} = M^2$$

where the term S_{d-1} arises immediately since the integrand only depends on the radial component

and hence is independent of the angular terms. We are left computing the radial term to then immediately obtain Ψ.

For this we compute using standard results for moments of normal distributions

$$\begin{aligned}\int_0^\infty \frac{1}{(2\pi\sigma^2)^{d/2}} r^{2k} e^{-\frac{r^2}{2\sigma^2}} r^{d-1} dr &= \frac{1}{(2\pi\sigma^2)^{(d-1)/2}} \int_0^\infty \frac{1}{(2\pi\sigma^2)^{1/2}} r^{2k+d-1} e^{-\frac{r^2}{2\sigma^2}} dr \\ &= \frac{1}{(2\pi\sigma^2)^{(d-1)/2}} \sigma^{2k+d-1} 2^{(2k+d-1)/2} \frac{\Gamma(\frac{2k+d}{2})}{\sqrt{2\pi}} \\ &= \sigma^{2k} 2^{k-1} \frac{1}{\pi^{d/2}} \Gamma(\frac{2k+d}{2}).\end{aligned}$$

The final result is a product of this and S_{d-1} which we can further simplify by using Lemma 88

$$\begin{aligned}S_{d-1}\sigma^{2k}2^{k-1}\frac{1}{\pi^{d/2}}\Gamma(\frac{2k+d}{2}) &= \sigma^{2k}2^k \frac{\Gamma(\frac{2k+d}{2})}{\Gamma(\frac{d}{2})} \\ &= \sigma^{2k}(d(d+2)\cdots(d+2(k-1))).\end{aligned}$$

We then have

$$\Psi = \frac{M^2}{\sigma^{2k} d(d+2)\cdots(d+2(k-1)}$$

and plugging in $\sigma = \frac{1}{\sqrt{d}}$ yields the result. □

Lemma 90. *Let p be a homogeneous polynomial of order 3 and let it have the property that $\mathbb{V}_{X,P}[P(X)] = M^2$ and that $P(X) \sim P(x')$ where $\|x\| = \|x'\|$ then we have*

$$\begin{aligned}\mathbb{V}_{X,P}[P_{a,V}(X)] &= \frac{M^2}{d(d+2)(d+4)} \left(15a^6 + 9a^4(d-1) + 3a^2(d-1)(d+1) + (d-1)(d+1)(d+3)\right) \\ &\leq M^2 \left(\frac{15a^6}{d^3} + \frac{9a^4}{d^2} + \frac{3a^2}{d} + 1\right).\end{aligned}$$

Proof. Let $X = \|X\|\hat{X}$ and by law of total variance and linearity of $T_{a,V}$ we have

$$\begin{aligned}\mathbb{V}_{C,X,V}[P_{a,V}(X)] &= \mathbb{E}_X[\mathbb{V}_P[P_{a,V}(X)]] \\ &= \mathbb{E}_X[\|T_{a,V}(\hat{X})\|^6 \ \|X\|^6 \ \mathbb{V}_P[P\left(\frac{T_{a,V}(\hat{X})}{\|T_{a,V}(\hat{X})\|}\right)]].\end{aligned}$$

We have by assumption that for any y and y' such that $\|y\| = \|y'\|$ we have that $P(y) \sim P(y')$. Since $\frac{T_{a,V}(\hat{X})}{\|T_{a,V}(\hat{X})\|}$ is a unit-vector we have that $\mathbb{V}_P[P\left(\frac{T_{a,V}(\hat{X})}{\|T_{a,V}(\hat{X})\|}\right)]$ is a constant which we denote by Ψ. In Lemma 79 we provide a closed-form expression for this quantity.

Next, we observe that $\mathbb{E}_{X,V}[\|T_{a,V}(\hat{X})\|^6 \ \|X\|^3 \ \Psi] = \mathbb{E}_X[\|T_{a,e_1}(\hat{X})\|^6 \ \|X\|^3 \ \Psi]$ which follows from the distribution of X being rotationally invariant. Specifically, for any given v we can rotate the coordinate system such that v in the given basis is equal to e_1 without changing the distribution. Hence, we say that without loss of generality $v = e_1$.

We now write $\|T_{a,e_1}(\hat{x})\|^2$ in hyper-spherical coordinates. For this notice that $\|T_{a,e_1}(\hat{x})\|^2 = 1 + \hat{x_1}^2(a^2 - 1)$ by Lemma 67. By definition of the coordinate system and since $\hat{x}$ is a unit vector we then have $\|T_a(\hat{x})\|^2 = 1 + \cos\phi_1{}^2(a^2 - 1) =: \|T_a(\phi_1)\|^2$.

We are now ready to write the expectation as an appropriate integral and decompose it into three independent components

$$\begin{aligned}&\mathbb{E}_X[\|T_{a,e_1}(\hat{X})\|^6 \ \|X\|^6 \ \Psi] \\ &= \int_V \|T_{a,v_1}(\hat{x})\|^6 \|x\|^6 \frac{\Psi}{(2\pi\sigma^2)^{d/2}} e^{-\frac{\|x\|^2}{2\sigma^2}} dV \\ &= \int_0^{2\pi}\int_0^{\pi}\cdots\int_0^{\infty} \|T_a(\phi_1)\|^6 \frac{\Psi}{(2\pi\sigma^2)^{d/2}} r^6 e^{-\frac{r^2}{2\sigma^2}} r^{d-1} \sin^{d-2}\phi_1 \cdots \sin\phi_{d-2} dr d\phi_1 \cdots d\phi_{d-1} \\ &= \left(\int_0^{\pi} \|T_a(\phi_1)\|^6 \sin^{d-2}\phi_1 d\phi\right)\left(\int_0^{2\pi}\cdots\int_0^{\pi} \sin^{d-3}\phi_2 \cdots \sin\phi_{d-2} d\phi_2 \cdots d\phi_{d-1}\right) \\ &\quad \left(\int_0^{\infty} \frac{\Psi}{(2\pi\sigma^2)^{d/2}} r^6 e^{-\frac{r^2}{2\sigma^2}} r^{d-1} dr\right)\end{aligned}$$

The integral nicely decomposes into three terms. We immediately observe that the middle term is simply $S_{(d-1)-1}$. We now compute the remaining terms.

For the radial term, we use the observation that by construction of p we have that $\mathbb{V}_{X,P}[P(X)] = \mathbb{E}_X[\|X\|^3 \mathbb{V}_P[P(\hat{X})]] = \mathbb{E}_X[\|X\|^3 \Psi] = M^2$. Written in hyper-spherical coordinates we have

$$\begin{aligned}
&\int_0^{2\pi} \int_0^{\pi} \cdots \int_0^{\infty} \frac{\Psi}{(2\pi\sigma^2)^{d/2}} r^6 e^{-\frac{r^2}{2\sigma^2}} r^{d-1} \sin^{d-2}\phi_1 \cdots \sin\phi_{d-2} dr d\phi_1 \cdots d\phi_{d-1} \\
&= \left(\int_0^{\infty} \frac{\Psi}{(2\pi\sigma^2)^{d/2}} r^6 e^{-\frac{r^2}{2\sigma^2}} r^{d-1} dr \right) \left(\int_0^{2\pi} \cdots \int_0^{\pi} \sin^{d-2}\phi_1 \cdots \sin\phi_{d-2} d\phi_1 \cdots d\phi_{d-1} \right) \\
&= \left(\int_0^{\infty} \frac{\Psi}{(2\pi\sigma^2)^{d/2}} r^6 e^{-\frac{r^2}{2\sigma^2}} r^{d-1} dr \right) S_{d-1} \\
&= M^2.
\end{aligned}$$

Hence, the radial part is equal to $\frac{M^2}{S_{d-1}}$.

We are left with the first term where we use $1 - \sin^2\phi_1 = \cos^2\phi_1$ which yields

$$\begin{aligned}
&\int_0^{\pi} \|T_a(\phi_1)\|^6 \sin^{d-2}\phi_1 d\phi_1 \\
&= \int_0^{\pi} \left(1 + (1 - \sin^2\phi_1)(a^2 - 1)\right)^3 \sin^{d-2}\phi_1 d\phi_1 \\
&= \int_0^{\pi} \left(a^6 - 3a^4(a^2-1)\sin^2\phi_1 + 3a^2(a^2-1)^2 \sin^4\phi_1 - (a^2-1)^3 \sin^6\phi_1\right) \sin^{d-2}\phi_1 d\phi_1.
\end{aligned}$$

To evaluate this, we need an expression for $\int_0^{\pi} \sin^m(x) dx$. For this, we use the following

$$\begin{aligned}
S_{(m+2)-1} &= \int_0^{2\pi} \cdots \int_0^{\pi} \sin^{(m+2)-2}\phi_1 \cdots \sin\phi_{(m+2)-2} d\phi_1 \cdots d\phi_{(m+2)-1} \\
&= \int_0^{\pi} \sin^m \phi_1 d\phi_1 \; S_{(m+1)-1}
\end{aligned}$$

and hence we have $\int_0^{\pi} \sin^m(x) dx = \frac{S_{(m+2)-1}}{S_{(m+1)-1}}$.

Using this, we can now simplify the previous integral and get

$$\begin{aligned}
&\int_0^{\pi} \left(a^6 - 3a^4(a^2-1)\sin^2\phi_1 + 3a^2(a^2-1)^2 \sin^4\phi_1 - (a^2-1)^3 \sin^6\phi_1\right) \sin^{d-2}\phi_1 d\phi_1 \\
&= a^6 \frac{S_{d-1}}{S_{(d-1)-1}} - 3a^4(a^2-1)\frac{S_{(d+2)-1}}{S_{(d+1)-1}} + 3a^2(a^2-1)^2 \frac{S_{(d+4)-1}}{S_{(d+3)-1}} - (a^2-1)^3 \frac{S_{(d+6)-1}}{S_{(d+5)-1}}.
\end{aligned}$$

Notably, to obtain the final result, we will multiply the above integral by $\frac{S_{(d-1)-1}}{S_{d-1}}$ arising from

the other two integrals.

Using Lemma 88 on the resulting ratios we have $c_1 = \frac{S_{(d+2)-1}}{S_{(d+1)-1}}\frac{S_{(d-1)-1}}{S_{d-1}} = \frac{d-1}{d}$ and $c_2 = \frac{S_{(d+4)-1}}{S_{(d+3)-1}}\frac{S_{(d-1)-1}}{S_{d-1}} = \frac{(d-1)(d+1)}{d(d+2)}$ and $c_3 = \frac{S_{(d+6)-1}}{S_{(d+5)-1}}\frac{S_{(d-1)-1}}{S_{d-1}} = \frac{(d-1)(d+1)(d+3)}{d(d+2)(d+4)}$. We use this to succinctly group together all orders of a as in the following and simply the resulting coefficients

$$\begin{aligned}
&a^6 - 3a^4(a^2-1)c_1 + 3a^2(a^2-1)^2c_2 - (a^2-1)^3c_3 \\
&\quad = a^6(1 - 3c_1 + 3c_2 - c_3) + a^4(3c_1 - 6c_2 + 3c_3) + a^2(3c_2 - 3c_3) + c_3 \\
&\quad = \frac{1}{d(d+2)(d+4)}\left(15a^6 + 9a^4(d-1) + 3a^2(d-1)(d+1) + (d-1)(d+1)(d+3)\right).
\end{aligned}$$

Multiplying by M^2 we have the result

$$\mathbb{V}_{X,P}[P_{a,V}(X)] = \frac{M^2}{d(d+2)(d+4)}\left(15a^6 + 9a^4(d-1) + 3a^2(d-1)(d+1) + (d-1)(d+1)(d+3)\right).$$

□

4.9 Summary

In this Chapter we provided a comprehensive study of the sample set for the centered simplex gradient. As claimed in our contributions, in Section 4.4.2 we presented Theorem 75 providing a novel result on the structure of odd homogeneous polynomials and a first link to the optimality of the sample set. In the same section, we introduced `ORCAS` and prove a range of worst-case guarantees. For one, we showed that with increasing dimension we approach best-case performance. Further, we provide Big-Theta rates on the performance of `ORCAS` compared to standard central differences—in the process indicating order of magnitude performance gains depending on function structure. In Section 4.5.2 we showed that `ORCAS` can be used as a dimensionality reduction method used in combination with numerical solvers. Without the use of `ORCAS` the numerical solver would be prohibitively slow. The benefit of this method is due to it allowing the last few sample points to break the orthogonality constraint, leading to a

potentially lower approximation error. In Sections 4.6 and 4.7 we presented extensive numerical results proving both theoretical and practical viability of our methodologies. Depending on the function structure, we show order of magnitude improvement over the current gold standard method. Hence, without increasing the cost of gradient estimation, we are able demonstrate orders of magnitude improvement in performance, implying that adapting insights and results from our work can significantly improve upon current methodologies. A key take-away of the study is that unlike the fully-determined simplex gradient, we have found that *orthogonal sample sets are often preferred*. This has to do with our theoretical proof of the ubiquity of orthogonal roots of third-order homogeneous polynomials. This result translates to being able to find many directions in which the approximation error, as discussed in the background Section 2.2.3, changes little and hence the difference vector lengths can be increased to mitigate the noise error.

Chapter 5

Conclusions

5.1 Summary of Thesis Achievements

In this thesis we have set out to study the sample set, that is, the location at which to evaluate a black-box system, as part of gradient estimation. Due to the prevalence of gradient estimation in a range of fields and applications, ranging from sensitivity analysis on parameters of biological processes [QM20, JEB+07, DPV06] to trajectory control and drone swarms [LK17, MCDDK16, KBD04], understanding how to reduce cost and improve accuracy can have a wide-ranging impact.

We specifically studied the sample set of the simplex gradient, based on a best linear estimator of the function. Here, we tackled how to optimally distribute the sample locations in the context of the approximation and noise error trade-off. We considered locations at which the approximation error is reduced, such that the distance between sample points can be increased in order to mitigate the noise error.

In Chapter 3 we focused on the fully-determined simplex gradient, a setting where the sample points are maximally reduced. We derived the optimal sample set for a quadratic function, which represents the best lower-order approximation of the function locally. We showed that unlike conventional wisdom, the sample set should often not be orthogonal, in order to *exploit low*

curvature directions. A crucial insight that may guide future gradient estimation developments. We validated these findings in practical settings via our historically informed global model framework, showing strong improvements. From these findings, we found that in many settings *our methodology can replace central difference, the gold standard gradient estimator, for half the cost.*

In Chapter 4, we provide a comprehensive study of the sample set for the centered simplex gradient. A gradient estimator that uses $2d$ sample points and a symmetric geometry. At a higher expense, this gradient estimator removes the lower-order approximation error which was considered in the previous chapter. We showed that the problem can be reduced to finding d orthogonal roots for a third order homogeneous polynomial. We then provided a constructive proof for the existence of $d-1$ orthogonal roots for any odd homogeneous polynomial, including third order polynomials. We showed that this is tight, as we can always construct polynomials for which there does not exist a complete set of d orthogonal roots. We then introduced `ORCAS`, a gradient estimation method and a dimensionality reduction method to be used in conjunction with a numerical solver. For the methodology, we provided extensive theoretical results that evaluated its performance, proving that *in higher dimensions it approaches the best-case performance and that has significantly improved worst-case guarantees than contemporary methods.* We validate these results with empirical experiments, for which we developed a polynomial sampling strategy. Finally, we prove the practical viability of our methods in the context of our historically informed global model framework. Overall, *we show that our methods significantly improve performance, often order of magnitude, for no additional cost.*

In summary, we have made several advances in the study of the optimal sample set for the simplex gradient. We have provided a study of the structure of the sample set and provided a first set of methodologies based on these results. These results have empirically been shown to be highly promising in improving the accuracy of gradient estimation methods without increasing cost.

5.2 Applications

We have discussed both applications of gradient estimation in sensitivity and derivative free optimization applications of gradient estimation in Section 2.4. Throughout the thesis, we have also hinted towards implicit definitions of gradient estimation which we have found during our studies. We want to briefly lay these out to demonstrate further potential use-cases of our techniques. These connections are specifically interesting, as they imply that many novel methods may have implicitly converged to use gradient estimation methods, specifically the simplex gradient, even though they were not explicitly designed to do so. By better understanding sample set properties and the connection to various fields, we may be able to unify many methods and develop a systematic framework in future works.

5.2.1 Interacting Particle Methods

Interacting particle methods are commonly designed to sample from complex high-dimensional probability distributions. This is done by using a large cloud of random samples termed particles. Practically, these particles evolve randomly around the space via the use of a potential function [DMD14]. Interacting particle methods have a range of applications from tracking and swarm optimization for trajectory planing [LK17, KBD04, MCDDK16], combinatorial counting [AV94], molecular and polymer simulation [Gra97], rare events simulation [CDMG11, GHSZ99] to quantum Monte Carlo methods [ACK00, Het84, MAN99].

As discussed in the context of derivative free optimization, in many situations where positions and particles are iteratively evolved a gradient is involved to compute the steepest descent update step. So, it may not be surprising that there is a strong connection to particle methods. In some cases the connection is laid out more clearly, such as for the Inverse Kalman Filter or general interacting particle based estimation and optimization use cases [CCSA20, GIHLS20, MTV15, WVDM00, WSG+14]. Another clear link has been made by Schillings to demonstrate the use of particle methods for ensemble-based gradient inference [STW23]. Then for an example where the gradient is not as clearly spelled out is for iterated filtering [IBAK11],

where the update step can be considered as coming from a likelihood ratio gradient estimator which has been touched upon by Doucet et al. [DJR13]. Lastly, Ollivier has pioneered the connection of many methods to natural gradient methods. These connections touch on Kalman Filters [Oll18, Oll19] and methods such as the CMA algorithm and natural evolution strategies [OAAH17, WSG+14, Han16].

5.2.2 Function Approximation

Another area where the methods may be useful is in function approximation. Piecewise linear functions are currently a focal point in research as many deep learning architectures represent this function class [PV18, TLH+22, PPL] and are trying to understand it [LPF+20]. Other application domains include modeling biological responses [Vie89], control allocation problems [BD04] and for problem formulations in optimization problems used for problem such as inventory allocation [MF10, AJD16]. The prelevance of this function class, seems to suggest that it has strong robustness and representation properties. Similar to how we described the benefit of using a local method to approximate gradients.

The importance of the function class has also led to a range of research. The difficulty in the piecewise linear problem is the selection of the interpolation points and the triangulation. There has been a focus on faster triangulation [CWL+18]. Other work then focuses on good selection of sample points such that the function is well represented [TLH+22, RK20, CN+15, LB22].

We hope that our methods can be applied in these settings as well. Each linear function in the piecewise linear functions can be seen as a locally best linear approximation. This is reminiscent of the gradient estimation problem. Here, we want to fit the gradient, which can in many instances also be seen as a local best linear approximation. Using our understanding of a good sample set, we may be able to better inform or understand the choice of interpolation points. We think this may be pretty interesting to explore in connection with deep learning methods, which do this implicitly via optimization. The networks show a strong approximation performance even for difficult problems [Yar18, ZCZG20, KSH12], implying that something interesting is happening, even though it had been a very difficult problem before.

5.3 Future Work

Given our work in the thesis, we see a range of potential future work. We divided the work into three core areas. One is to use our insights and methodologies in the context of the discussed application domains. The other is a continued study of approximating function dependent information to inform sample set selection. The last is to further study the geometry of the sample set for the simplex gradient. We briefly expand on each in the following.

5.3.1 Applications

As we have seen throughout the thesis, there is a range of domains in which gradient estimation is used. In this work, we have mainly worked in an idealized setting and studied the properties of the sample set. While our applications in Derivative Free Optimization and Sensitivity Analysis in Section 3.5 were insightful, there remain many more intricacies that can be explored and methods that can be developed which leverage domain knowledge. Future work would then included focusing on a specific application domain and use the insights gained to develop specialized methods. Such specialized methods then promise to further increase performance. Hence it is crucial to strengthen the connection between gradient estimation and a range of domains in which it may have only been implicitly deployed so far. To provide some starting points we present the following.

- Develop a more foundational connection to Design of Experiment (DoE) schemes as used for Sensitivity Analysis. In Chapter 3 we have developed the optimal sample set structure for the fully-determined simplex gradient which immediately leads to an interesting connection to the Plackett–Burman design in DoE [PB46, BDHH05] via the use of the Hadamard matrix. Understanding how the optimal structures which we have derived can be used to augmented domain specific designs may be promising.

- In derivative free optimization and interacting particle schemes, particles and positions are evolved iteratively according to some update rule. When a gradient estimate is used

to inform the update, we naturally want that the particles which inform this estimate are appropriately distributed to provide an accurate yet cheap gradient estimate. The natural constraint is that moving particles may be expensive and hence movements should be minimized [FZL+22, FTL+24]. A valuable study is to then understand how to evolve particles to retain a geometry which allows for accurate gradient estimates while adhering to any movement constraints.

- Finally, we would like to use our insights for the sample set selection to understand the selection of data for piecewise linear function approximation. As discussed above, this is an immensely important area due to the prevalence of such function approximation methods in the current deep learning age. By using the understanding that a gradient estimate is a best-linear fit, there is an immediate connection to our developed methods for the simplex gradient. We expect this to be an involved but highly rewarding study.

5.3.2 Historically Informed Sample Set Construction

To practically apply our methodologies, we require estimates of function dependent information. To address this in our work we have developed the historically informed global model framework to improve on other work done in this area [GMW19, GMSW83, SXXN22, MW12, Bar92]. However, we believe that there may be significantly improved methods to allow informed sample set selection methods.

We still do not know exactly when a global model estimate will be informative enough for improved sample set selection. For example, in Section 3.5.3, which addresses a derivative free optimization application, the use of the Hessian model from BFGS may be useful, we found that this is far inferior to using a global model as used in our framework. We do not yet know why this is true. Hence, a further study on the sensitivity of the sample set to various function dependent information estimates may provide a more complete understanding.

Overall, exploiting problem specific properties to design methods which allow for informed sample set selection is a promising avenue to produce improved methods in a range of application

domains. We hope that our insights will provide a better guideline for which estimates may be needed to construct such improved methods.

5.3.3 Further Study of Optimal Sample Sets

Lastly, there remain a range of directions for the study of optimal sample set. While we have studied the sample set for the two most commonly used instances of the simplex gradient, we have made some assumptions which may be relaxed. Furthermore, we believe that there is more room left to study the dependence of our methods on variations in the problem set-up, which may aid in developing more principled gradient estimation methods.

- In Chapter 3 we studied the fully-determined simplex gradient, but made the assumption that the point at which the gradient is to be evaluated is included in the sample set. We conjecture that relaxing this assumption could lead to a significant performance improvement. This relies on an intuitive understanding that for a positive definite quadratic function we can now draw a hyperplane which has the same slope as the gradient. However, this best-fit function would not interpolate the location of gradient estimation.

- We believe that it would also be highly important to study the sensitivity of the sample set. Here, one may investigate what the sensitivity of varying the sample set slightly would be on the performance of the gradient estimation method. Then, one may study the sensitivity of the sample set on the accuracy of estimating the noise level and the approximation error. Understanding these sensitivities will significantly help with gradient estimation development. One will want to avoid sample sets with high-sensitivity, as then a method may overfit to an inaccurate estimate of either the approximation error or noise level and hence lead to poor performance.

- In Chapter 4 we used a numerical method to find the last few sample points. While this appears to work in our extensive numerical results and provide near optimal performance,

one may aim to devise a method which does not rely on a numerical solver and directly provides good sample point locations.

- In all of our studies we assume constant noise level. While for many settings this is appropriate, relaxing this assumption will lead to significantly increased generality. Here, considering settings where functions are drawn from Gaussian Processes may be of great interest, as they provide additional function structure which may be exploited [MK22].

- Lastly, we only considered a mean squared error function. However, there are a range of other metrics which may be used to assess the accuracy of the gradient estimate. One interesting extension may be to consider the cosine similarity. Here, all the weight is placed on obtaining the right direction. Specifically in a derivative free optimization setting, where line-search methods are deployed, this may significantly improve the performance of the gradient estimate.

Bibliography

[AA07] Karsten Ahnert and Markus Abel. Numerical differentiation of experimental data: local versus global methods. *Computer Physics Communications*, 177(10):764–774, 2007.

[ABAFS18] Mohamed Abdel-Basset, Laila Abdel-Fatah, and Arun Kumar Sangaiah. Metaheuristic algorithms: A comprehensive review. *Computational intelligence for multimedia big data on the cloud with engineering applications*, pages 185–231, 2018.

[ACK00] Roland Assaraf, Michel Caffarel, and Anatole Khelif. Diffusion monte carlo methods with a fixed number of walkers. *Physical Review E*, 61(4):4566, 2000.

[AH17] Charles Audet and Warren Hare. Derivative-free and blackbox optimization. 2017.

[AJD16] Amir Ardestani-Jaafari and Erick Delage. Robust optimization of sums of piecewise linear functions with application to inventory problems. *Operations research*, 64(2):474–494, 2016.

[AV94] David Aldous and Umesh Vazirani. " go with the winners" algorithms. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 492–501. IEEE, 1994.

[Bar92] Russell R Barton. Computing forward difference derivatives in engineering optimization. *Engineering optimization*, 20(3):205–224, 1992.

[BBN19] Albert S Berahas, Richard H Byrd, and Jorge Nocedal. Derivative-free optimization

of noisy functions via quasi-newton methods. *SIAM Journal on Optimization*, 29(2):965–993, 2019.

[BCCS19] Albert S Berahas, Liyuan Cao, Krzysztof Choromanski, and Katya Scheinberg. Linear interpolation gives better gradients than gaussian smoothing in derivative-free optimization. *arXiv preprint arXiv:1905.13043*, 2019.

[BCCS22] Albert S Berahas, Liyuan Cao, Krzysztof Choromanski, and Katya Scheinberg. A theoretical and empirical comparison of gradient approximations in derivative-free optimization. *Foundations of Computational Mathematics*, 22(2):507–560, 2022.

[BD04] Michael A Bolender and David B Doman. Nonlinear control allocation using piecewise linear functions. *Journal of Guidance, Control, and Dynamics*, 27(6):1017–1027, 2004.

[BDHdH08] RCM Brekelmans, LT Driessen, HJM Hamers, and Dick den Hertog. Gradient estimation using lagrange interpolation polynomials. *Journal of optimization theory and applications*, 136(3):341–357, 2008.

[BDHH05] RCM Brekelmans, LT Driessen, HJM Hamers, and D den Hertog. Gradient estimation schemes for noisy functions. *Journal of Optimization Theory and Applications*, 126(3):529–551, 2005.

[BDS17] Vivek S Borkar, Vikranth R Dwaracherla, and Neeraja Sahasrabudhe. Gradient estimation with simultaneous perturbation and compressive sensing. *The Journal of Machine Learning Research*, 18(1):5910–5936, 2017.

[BFB15] Richard L Burden, J Douglas Faires, and Annette M Burden. *Numerical analysis*. Cengage learning, 2015.

[BH12] O Bruno and D Hoch. Numerical differentiation of approximated functions with limited order-of-accuracy deterioration. *SIAM Journal on Numerical Analysis*, 50(3):1581–1603, 2012.

[BK98] David Matthew Bortz and Carl Tim Kelley. The simplex gradient and noisy optimization problems. In *Computational Methods for Optimal Design and Control: Proceedings of the AFOSR Workshop on Optimal Design and Control Arlington, Virginia 30 September–3 October, 1997*, pages 77–90. Springer, 1998.

[BKK19] Sujay Bhatt, Alec Koppel, and Vikram Krishnamurthy. Policy gradient using weak derivatives for reinforcement learning. pages 5531–5537. IEEE, 2019 IEEE 58th Conference on Decision and Control (CDC), 2019.

[BKW24] Raghu Bollapragada, Cem Karamanli, and Stefan M Wild. Derivative-free optimization via adaptive sampling strategies. *arXiv preprint arXiv:2404.11893*, 2024.

[Bor22] Vivek S Borkar. Stochastic gradient schemes. In *Stochastic Approximation: A Dynamical Systems Viewpoint: Second Edition*, pages 191–225. Springer, 2022.

[BPL19] David Brookes, Hahnbeom Park, and Jennifer Listgarten. Conditioning by adaptive sampling for robust design. In *International conference on machine learning*, pages 773–782. PMLR, 2019.

[BPP13] S.. Bhatnagar, HL Prasad, and LA Prashanth. *Stochastic Recursive Algorithms for Optimization: Simultaneous Perturbation Methods*. Springer, 2013.

[BPRS18] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Marchine Learning Research*, 18:1–43, 2018.

[BZ19] Adam Blakney and Jingyi Zhu. A comparison of the finite difference and simultaneous perturbation gradient estimation methods with noisy function evaluations. pages 1–6. IEEE, 2019 53rd Annual Conference on Information Sciences and Systems (CISS), 2019.

[CCI16] Víctor Cerdà, Juan Luis Cerdà, and Abubakr M Idris. Optimization using the gradient and simplex methods. *Talanta*, 148:641–648, 2016.

[CCSA20] Neil K Chada, Yuming Chen, and Daniel Sanz-Alonso. Iterative ensemble kalman methods: A unified perspective with some new variants. *arXiv preprint arXiv:2010.13299*, 2020.

[CDMG11] Frédéric Cérou, Pierre Del Moral, and Arnaud Guyader. A nonasymptotic theorem for unnormalized feynman-kac particle models. In *Annales de l'IHP Probabilités et statistiques*, volume 47, pages 629–649, 2011.

[CDV08] AL Custódio, JE Dennis, and Luís Nunes Vicente. Using simplex gradients of nonsmooth functions in direct search methods. *IMA journal of numerical analysis*, 28(4):770–784, 2008.

[CH23] Yiwen Chen and Warren Hare. Adapting the centred simplex gradient to compensate for misaligned sample points. *IMA Journal of Numerical Analysis*, page drad050, 2023.

[CHJB24] Yiwen Chen, Warren Hare, and Gabriel Jarry-Bolduc. Error analysis of surrogate models constructed through operations on submodels. *Mathematics of Operations Research*, 49(1):1–18, 2024.

[CL09] E.W. Cheney and W.A. Light. *A Course in Approximation Theory*. Graduate studies in mathematics. American Mathematical Society, 2009.

[CN+15] Eduardo Camponogara, Luiz Fernando Nazari, et al. Models and algorithms for optimal piecewise-linear function approximation. *Mathematical Problems in Engineering*, 2015, 2015.

[CSV08a] Andrew R Conn, Katya Scheinberg, and Luís N Vicente. Geometry of interpolation sets in derivative free optimization. *Mathematical programming*, 111:141–172, 2008.

[CSV08b] Andrew R Conn, Katya Scheinberg, and Luis N Vicente. Geometry of sample sets in derivative-free optimization: polynomial regression and underdetermined interpolation. *IMA journal of numerical analysis*, 28(4):721–748, 2008.

[CT06] Gerard Cornuejols and Reha Tütüncü. *Optimization methods in finance*, volume 5. Cambridge University Press, 2006.

[CT19] Ian Coope and Rachael Tappenden. Efficient calculation of regular simplex gradients. *Computational Optimization and Applications*, 72:561–588, 2019.

[Cul71] Jane Cullum. Numerical differentiation and regularization. *SIAM Journal on numerical analysis*, 8(2):254–265, 1971.

[CV07] Ana Luísa Custódio and Luís Nunes Vicente. Using sampling and simplex derivatives in pattern search methods. *SIAM Journal on Optimization*, 18(2):537–555, 2007.

[CWL+18] Tyler H Chang, Layne T Watson, Thomas CH Lux, Bo Li, Li Xu, Ali R Butt, Kirk W Cameron, and Yili Hong. A polynomial time algorithm for multivariate interpolation in arbitrary dimension via the delaunay triangulation. In *Proceedings of the ACMSE 2018 Conference*, pages 1–8, 2018.

[DB03] Germund Dahlquist and Åke Björck. *Numerical methods*. Courier Corporation, 2003.

[DJR13] Arnaud Doucet, Pierre E Jacob, and Sylvain Rubenthaler. Derivative-free estimation of the score vector and observed information matrix with application to state-space models. *arXiv preprint arXiv:1304.5768*, 2013.

[DMD14] Piere Del Moral and Arnaud Doucet. Particle methods: An introduction with applications. In *ESAIM: proceedings*, volume 44, pages 1–46. EDP Sciences, 2014.

[DPV06] Dirk JW De Pauw and Peter A Vanrolleghem. Practical aspects of sensitivity function approximation for dynamic models. *Mathematical and Computer Modelling of Dynamical Systems*, 12(5):395–414, 2006.

[FCJR17] Rahul Rahul-Mark Fonseca, Bailian Chen, Jan Dirk Jansen, and Albert Reynolds. A stochastic simplex approximate gradient (stosag) for optimization under uncer-

tainty. *International Journal for Numerical Methods in Engineering*, 109(13):1756–1776, 2017.

[FFN21] Jan N Fuhg, Amélie Fau, and Udo Nackenhorst. State-of-the-art and comparative review of adaptive sampling methods for kriging. *Archives of Computational Methods in Engineering*, 28:2689–2747, 2021.

[FH75] Keinosuke Fukunaga and Larry Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, 21(1):32–40, 1975.

[FH12] Michael C Fu and Jian-Qiang Hu. *Conditional Monte Carlo: Gradient estimation and optimization applications*, volume 392. Springer Science & Business Media, 2012.

[Fis36] Ronald Aylmer Fisher. Design of experiments. *British Medical Journal*, 1(3923):554, 1936.

[FKM04] Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. *arXiv preprint cs/0408007*, 2004.

[For88] Bengt Fornberg. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of computation*, 51(184):699–706, 1988.

[FTL+24] Jose Pablo Folch, Calvin Tsay, Robert M Lee, Behrang Shafei, Weronika Ormaniec, Andreas Krause, Mark van der Wilk, Ruth Misener, and Mojmír Mutnỳ. Transition constrained bayesian optimization via markov decision processes. *arXiv preprint arXiv:2402.08406*, 2024.

[Fu06] Michael C Fu. Gradient estimation. *Handbooks in operations research and management science*, 13:575–616, 2006.

[FZL+22] Jose Pablo Folch, Shiqiang Zhang, Robert Lee, Behrang Shafei, David Walz, Calvin Tsay, Mark van der Wilk, and Ruth Misener. Snake: Bayesian optimization

with pathwise exploration. *Advances in Neural Information Processing Systems*, 35:35226–35239, 2022.

[G+89] Andreas Griewank et al. On automatic differentiation. *Mathematical Programming: recent developments and applications*, 6(6):83–107, 1989.

[GHSZ99] Paul Glasserman, Philip Heidelberger, Perwez Shahabuddin, and Tim Zajic. Multilevel splitting for estimating rare event probabilities. *Operations Research*, 47(4):585–600, 1999.

[GIHLS20] Alfredo Garbuno-Inigo, Franca Hoffmann, Wuchen Li, and Andrew M Stuart. Interacting langevin diffusions: Gradient structure and ensemble kalman sampler. *SIAM Journal on Applied Dynamical Systems*, 19(1):412–441, 2020.

[GLS00] Christian Gourieroux, Jean-Paul Laurent, and Olivier Scaillet. Sensitivity analysis of values at risk. *Journal of empirical finance*, 7(3-4):225–245, 2000.

[Gly90] Peter W Glynn. Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84, 1990.

[GMS19] Manfred Gilli, Dietmar Maringer, and Enrico Schumann. *Numerical methods and optimization in finance*. Academic Press, 2019.

[GMSW83] Philip E. Gill, Walter Murray, Michael A. Saunders, and Margaret H. Wright. Computing forward-difference intervals for numerical optimization. *SIAM Journal on Scientific and Statistical Computing*, 4(2):310–321, 1983.

[GMW19] Philip E Gill, Walter Murray, and Margaret H Wright. *Practical optimization*. SIAM, 2019.

[GOT15] Nicholas IM Gould, Dominique Orban, and Philippe L Toint. Cutest: a constrained and unconstrained testing environment with safe threads for mathematical optimization. *Computational optimization and applications*, 60:545–557, 2015.

[Gra97] Peter Grassberger. Pruned-enriched rosenbluth method: Simulations of θ polymers of chain length up to 1 000 000. *Physical Review E*, 56(3):3682, 1997.

[Han16] Nikolaus Hansen. The cma evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*, 2016.

[Har13] Robin Hartshorne. *Algebraic geometry*, volume 52. Springer Science & Business Media, 2013.

[Het84] Jack H Hetherington. Observations on the statistical iteration of matrices. *Physical Review A*, 30(5):2713, 1984.

[HJB20] Warren Hare and Gabriel Jarry-Bolduc. Calculus identities for generalized simplex gradients: Rules and applications. *SIAM Journal on Optimization*, 30(1):853–884, 2020.

[HJBP22] Warren Hare, Gabriel Jarry-Bolduc, and Chayne Planiden. Error bounds for overdetermined and underdetermined generalized centred simplex gradients. *IMA Journal of Numerical Analysis*, 42(1):744–770, 2022.

[HRR23] Warren Hare, Lindon Roberts, and Clément W Royer. Expected decrease for derivative-free algorithms using random subspaces. *arXiv preprint arXiv:2308.04734*, 2023.

[HS01] Martin Hanke and Otmar Scherzer. Inverse problems light: numerical differentiation. *The American Mathematical Monthly*, 108(6):512–521, 2001.

[IBAK11] Edward L Ionides, Anindya Bhadra, Yves Atchadé, and Aaron King. Iterated filtering. 2011.

[IL15] Bertrand Iooss and Paul Lemaître. A review on global sensitivity analysis methods. *Uncertainty management in simulation-optimization of complex systems: algorithms and applications*, pages 101–122, 2015.

[JEB+07] Matthew D Johnston, Carina M Edwards, Walter F Bodmer, Philip K Maini, and S Jonathan Chapman. Examples of mathematical modeling: tales from the crypt. *Cell cycle*, 6(17):2106–2112, 2007.

[KBD04] Zia Khan, Tucker Balch, and Frank Dellaert. An mcmc-based particle filter for tracking multiple interacting targets. In *Computer Vision-ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part IV 8*, pages 279–290. Springer, 2004.

[Kel11] Carl T Kelley. *Implicit filtering*. SIAM, 2011.

[Ken05] David A Kendrick. Stochastic control for economic models: past, present and the paths ahead. *Journal of economic dynamics and control*, 29(1-2):170–171, 2005.

[KIT+19] Ingrid Kovacs, Alexandra Iosub, Marina Topa, Andi Buzo, and Georg Pelz. A gradient-based sensitivity analysis method for complex systems. In *2019 IEEE 25th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, pages 333–338. IEEE, 2019.

[KO72] V Ya Katkovnik and KULCHITS. OY. Convergence of a class of random search algorithms. *Automation and Remote Control*, 33(8):1321–1326, 1972.

[KR14] Ian Knowles and Robert J Renka. Methods for numerical differentiation of noisy data. *Electron. J. Differ. Equ*, 21:235–246, 2014.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[KTR05] Hadi Kharaghani and Behruz Tayfeh-Rezaie. A hadamard matrix of order 428. *Journal of Combinatorial Designs*, 13(6):435–440, 2005.

[LB22] Daniel Lengyel and Anastasia Borovykh. Efficient regression with deep neural networks: how many datapoints do we need? In *Has it Trained Yet? NeurIPS 2022 Workshop*, 2022.

[LBXY19] Turab Lookman, Prasanna V Balachandran, Dezhen Xue, and Ruihao Yuan. Active learning in materials science with emphasis on adaptive sampling using uncertainties for targeted design. *npj Computational Materials*, 5(1):21, 2019.

[LK17] Hyeonbeom Lee and H Jin Kim. Trajectory tracking control of multirotors from modelling to experiments: A survey. *International Journal of Control, Automation and Systems*, 15(1):281–292, 2017.

[LMW19] Jeffrey Larson, Matt Menickelly, and Stefan M Wild. Derivative-free optimization methods. *Acta Numerica*, 28:287–404, 2019.

[LOC18] Haitao Liu, Yew-Soon Ong, and Jianfei Cai. A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design. *Structural and Multidisciplinary Optimization*, 57:393–416, 2018.

[LPF+20] Daniel Lengyel, Janith Petangoda, Isak Falk, Kate Highnam, Michalis Lazarou, Arinbjörn Kolbeinsson, Marc Peter Deisenroth, and Nicholas R Jennings. Genni: Visualising the geometry of equivalences for neural network identifiability. *arXiv preprint arXiv:2011.07407*, 2020.

[LPKJ23] Daniel Lengyel, Panos Parpas, Nikolas Kantas, and Nicholas R Jennings. Curvature aligned simplex gradient: Principled sample set construction for numerical differentiation. *arXiv preprint arXiv:2310.12712*, 2023.

[MAN99] Vilen Melik-Alaverdian and MP Nightingale. Quantum monte carlo methods in statistical mechanics. *International Journal of Modern Physics C*, 10(08):1409–1418, 1999.

[MCDDK16] Thi Thoa Mac, Cosmin Copot, Trung Tran Duc, and Robin De Keyser. Ar. drone uav control parameters tuning based on particle swarm optimization algorithm. In *2016 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, pages 1–6. IEEE, 2016.

[MF10] R Misener and CA Floudas. Piecewise-linear approximations of multidimensional functions. *Journal of optimization theory and applications*, 145:120–147, 2010.

[MFT08] Alison L Marsden, Jeffrey A Feinstein, and Charles A Taylor. A computational framework for derivative-free optimization of cardiovascular geometries. *Computer methods in applied mechanics and engineering*, 197(21-24):1890–1905, 2008.

[MK22] Mojmir Mutny and Andreas Krause. Experimental design for linear functionals in reproducing kernel hilbert spaces. *Advances in Neural Information Processing Systems*, 35:20175–20188, 2022.

[Mon17] Douglas C Montgomery. *Design and analysis of experiments*. John wiley & sons, 2017.

[MOV12] Jacob A McGill, Babatunde A Ogunnaike, and Dionisios G Vlachos. Efficient gradient estimation using finite differencing and likelihood ratios for kinetic monte carlo simulations. *Journal of Computational Physics*, 231(21):7170–7186, 2012.

[MRFM20] Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte carlo gradient estimation in machine learning. *Journal of Machine Learning Research*, 21(132):1–62, 2020.

[MTV15] Arrchana Muruganantham, Kay Chen Tan, and Prahlad Vadakkepat. Evolutionary dynamic multiobjective optimization via kalman filter prediction. *IEEE transactions on cybernetics*, 46(12):2862–2873, 2015.

[MW09] Jorge J Moré and Stefan M Wild. Benchmarking derivative-free optimization algorithms. *SIAM Journal on Optimization*, 20(1):172–191, 2009.

[MW12] Jorge J Moré and Stefan M Wild. Estimating derivatives of noisy simulations. *ACM Transactions on Mathematical Software (TOMS)*, 38(3):1–21, 2012.

[MWDS18] Alvaro Maggiar, Andreas Wachter, Irina S Dolinskaya, and Jeremy Staum. A derivative-free trust-region algorithm for the optimization of functions smoothed via gaussian convolution using adaptive multiple importance sampling. *SIAM Journal on Optimization*, 28(2):1478–1507, 2018.

[NM65] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.

[NW06] Jorge Nocedal and Stephen J Wright. Line search methods. *Numerical optimization*, pages 30–65, 2006.

[OAAH17] Yann Ollivier, Ludovic Arnold, Anne Auger, and Nikolaus Hansen. Information-geometric optimization algorithms: A unifying picture via invariance principles. *Journal of Machine Learning Research*, 18(18):1–65, 2017.

[Oll18] Yann Ollivier. Online natural gradient as a kalman filter. 2018.

[Oll19] Yann Ollivier. The extended kalman filter is a natural gradient descent in trajectory space. *arXiv preprint arXiv:1901.00696*, 2019.

[Pan97] David J Pannell. Sensitivity analysis of normative economic models: theoretical framework and practical strategies. *Agricultural economics*, 16(2):139–152, 1997.

[PB46] Robin L Plackett and J Peter Burman. The design of optimum multifactorial experiments. *Biometrika*, 33(4):305–325, 1946.

[PPL] Rahil Pandya, Panos Parpas, and Daniel Lengyel. From plateaus to progress: Unveiling training dynamics of pinns.

[PV18] Philipp Petersen and Felix Voigtlaender. Optimal approximation of piecewise smooth functions using deep relu neural networks. *Neural Networks*, 108:296–330, 2018.

[QM20] George Qian and Adam Mahdi. Sensitivity analysis methods in the biomedical sciences. *Mathematical biosciences*, 323:108306, 2020.

[RB16] Arunselvan Ramaswamy and Shalabh Bhatnagar. Stochastic recursive inclusion in two timescales with an application to the lagrangian dual problem. *Stochastics*, 88(8):1173–1187, 2016.

[Reg15] Rommel G Regis. The calculus of simplex gradients. *Optimization Letters*, 9(5):845–865, 2015.

[RK16] Reuven Y Rubinstein and Dirk P Kroese. *Simulation and the Monte Carlo method*. John Wiley & Sons, 2016.

[RK20] Steffen Rebennack and Vitaliy Krasko. Piecewise linear function fitting via mixed-integer linear programming. *INFORMS Journal on Computing*, 32(2):507–530, 2020.

[S+92] James C Spall et al. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control*, 37(3):332–341, 1992.

[SB16] Marie-Francoise Roy Saugata Basu, Richard Pollack. Algorithms in real algebraic geometry. 2016.

[SBK+20] Sergey Shirobokov, Vladislav Belavin, Michael Kagan, Andrei Ustyuzhanin, and Atilim Gunes Baydin. Black-box optimization with local generative surrogates. *Advances in Neural Information Processing Systems*, 33:14650–14662, 2020.

[Sch22] Katya Scheinberg. Finite difference gradient approximation: To randomize or not? *INFORMS Journal on Computing*, 2022.

[SHWA15] John Schulman, Nicolas Heess, Theophane Weber, and Pieter Abbeel. Gradient estimation using stochastic computation graphs. *Advances in Neural Information Processing Systems*, 28, 2015.

[Sob09] Eric A Sobie. Parameter sensitivity analysis in electrophysiological models using multivariable regression. *Biophysical journal*, 96(4):1264–1274, 2009.

[Spa98] James C Spall. An overview of the simultaneous perturbation method for efficient optimization. *Johns Hopkins apl technical digest*, 19(4):482–492, 1998.

[SPD18] Christoph Schied, Christoph Peters, and Carsten Dachsbacher. Gradient estimation for real-time adaptive temporal filtering. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 1(2):1–16, 2018.

[SQXON23] Hao-Jun Michael Shi, Melody Qiming Xuan, Figen Oztoprak, and Jorge Nocedal. On the numerical performance of finite-difference-based methods for derivative-free optimization. *Optimization Methods and Software*, 38(2):289–311, 2023.

[SSSC20] Yijia Sun, Nikolaos V Sahinidis, Anantha Sundaram, and Myun-Seok Cheon. Derivative-free optimization for chemical product design. *Current Opinion in Chemical Engineering*, 27:98–106, 2020.

[STW23] Claudia Schillings, Claudia Totzeck, and Philipp Wacker. Ensemble-based gradient inference for particle methods in optimization and sampling. *SIAM/ASA Journal on Uncertainty Quantification*, 11(3):757–787, 2023.

[Sur87] Rajan Suri. Infinitesimal perturbation analysis for general discrete event systems. *Journal of the ACM (JACM)*, 34(3):686–717, 1987.

[SXXN22] Hao-Jun Michael Shi, Yuchen Xie, Melody Qiming Xuan, and Jorge Nocedal. Adaptive finite-difference interval estimation for noisy derivative-free optimization. *SIAM Journal on Scientific Computing*, 44(4):A2302–A2321, 2022.

[SY06] Wenyu Sun and Ya-Xiang Yuan. *Optimization theory and methods: nonlinear programming*, volume 1. Springer Science & Business Media, 2006.

[TLH+22] Qinghua Tao, Li Li, Xiaolin Huang, Xiangming Xi, Shuning Wang, and Johan AK Suykens. Piecewise linear neural networks and deep learning. *Nature Reviews Methods Primers*, 2(1):42, 2022.

[TMZ+13] Lehana Thabane, Lawrence Mbuagbaw, Shiyuan Zhang, Zainab Samaan, Maura Marcucci, Chenglin Ye, Marroon Thabane, Lora Giangregorio, Brittany Dennis, Daisy Kosa, et al. A tutorial on sensitivity analyses in clinical trials: the what, why, when and how. *BMC medical research methodology*, 13:1–12, 2013.

[TS17] Seiya Tokui and Issei Sato. Evaluating the variance of likelihood-ratio gradient estimators. In *International Conference on Machine Learning*, pages 3414–3423. PMLR, 2017.

[VA00] Felisa J Vazquez-Abad. A course on sensitivity analysis for gradient estimation of des performance measures. In *Discrete Event Systems: Analysis and Control*, pages 3–28. Springer, 2000.

[VBKB20] Floris Van Breugel, J Nathan Kutz, and Bingni W Brunton. Numerical differentiation of noisy data: A unifying multi-objective optimization framework. *IEEE Access*, 8:196865–196877, 2020.

[VGO+20] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[Vie89] Elisabeth Vieth. Fitting piecewise linear regression functions to biological responses. *Journal of applied physiology*, 67(1):390–396, 1989.

[WCC18] Yorai Wardi, Christos G Cassandras, and Xi-Ren Cao. Perturbation analysis: a framework for data-driven control and optimization of discrete event and hybrid systems. *Annual Reviews in Control*, 45:267–280, 2018.

[WH07] Ting Wei and YC Hon. Numerical differentiation by radial basis functions approximation. *Advances in Computational Mathematics*, 27(3):247–272, 2007.

[Wil92] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

[WSG+14] Daan Wierstra, Tom Schaul, Tobias Glasmachers, Yi Sun, Jan Peters, and Jürgen Schmidhuber. Natural evolution strategies. *The Journal of Machine Learning Research*, 15(1):949–980, 2014.

[Wu96] Zhijun Wu. The effective energy transformation scheme as a special continuation approach to global optimization with application to molecular conformation. *SIAM Journal on Optimization*, 6(3):748–768, 1996.

[WVDM00] Eric A Wan and Rudolph Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 adaptive systems for signal processing, communications, and control symposium (Cat. No. 00EX373)*, pages 153–158. Ieee, 2000.

[Yar18] Dmitry Yarotsky. Optimal approximation of continuous functions by very deep relu networks. In *Conference on learning theory*, pages 639–649. PMLR, 2018.

[ZCZG20] Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Gradient descent optimizes over-parameterized deep relu networks. *Machine learning*, 109:467–492, 2020.

[ZS91] JC Zhang and MA Styblinski. Design of experiments approach to gradient estimation and its application to cmos circuit stochastic optimization. pages 3098–3101. IEEE, 1991., IEEE International Sympoisum on Circuits and Systems, 1991.