SIAM J. Applied Dynamical Systems
Vol. 17, No. 2, pp. 1432–1477

The Optimization Landscape for Fitting a Rank-2 Tensor with a Rank-1 Tensor*

Xue Gong†, Martin J. Mohlenkamp‡, and Todd R. Young‡

Abstract. The ability to approximate a multivariate function/tensor as a sum of separable functions/tensors is quite useful. Unfortunately, optimization-based algorithms to do so are not robust and regularly exhibit unusual transient behavior. A variety of different algorithms have been proposed with different features, but the state of the art is still unsatisfactory. In this work we step back from the algorithms and study the optimization problem that these algorithms are trying to solve. We apply dynamical systems concepts to analyze the simplest nontrivial case, which is a rank-2 tensor approximated by a rank-1 tensor. We find nonhyperbolic minima and saddles, which would be difficult for algorithms to handle. These features occur at relatively small but nonzero angles, rather than in the small-angle limit as the literature would suggest. We also identify transverse stability as a mechanism that may explain the slow convergence of these algorithms more generally.

Key words. tensor approximation, alternating least squares (ALS), canonical tensor format, swamps

AMS subject classifications. 15A69, 37N30, 65K10

DOI. 10.1137/17M112213X

1. Introduction.

Consider the problem of approximating a given target tensor T by another tensor G that is written as a (short) sum of separable tensors. We can write this problem with indexes as

$$T(j_1, j_2, \ldots, j_d) \approx G(j_1, j_2, \ldots, j_d) = \sum_{l=1}^{r} \prod_{i=1}^{d} G_i^l(j_i) \quad \text{for} \quad j_i = 1, 2, \ldots, M_i \tag{1}$$

or without indexes as

$$T \approx G = \sum_{l=1}^{r} \bigotimes_{i=1}^{d} G_i^l \,. \tag{2}$$

This sum-of-separable format for G is known variously as the canonical format, the canonical polyadic format, the CANDECOMP/PARAFAC format, or a separated representation (e.g., [9, 10, 21, 24, 49, 52, 56, 57, 68, 106]). Such an approximation has applications in data analysis and signal processing (e.g., [8, 15, 21, 25, 52, 68, 75, 76]), numerical methods in high dimensions (e.g., [4, 5, 9, 10, 11, 13, 50, 51, 63, 72, 73, 74, 85, 86, 97, 122, 130]), uncertainty quantification (e.g., [7, 23, 37, 65, 86, 97]), and others (e.g., [8, 29]). Alternative tensor formats exist and work well for some applications (e.g., [6, 46, 64, 87, 88, 89, 90, 99]), but the

*Received by the editors March 22, 2017; accepted for publication (in revised form) by D. Barkley January 12, 2018; published electronically May 17, 2018.
http://www.siam.org/journals/siads/17-2/M112213.html
Funding: This material is based upon work supported by the National Science Foundation under Grant 1418787.
†Department of Mathematics, Statistics and Computer Science, University of Wisconsin-Stout, Menomonie, WI 54751 (gongx@uwstout.edu).
‡Department of Mathematics, Ohio University, Athens, OH 45701 (mohlenka@ohio.edu, youngt@ohio.edu).

sum-of-separable tensor format remains important and appears to be the only format that can be used in certain applications, such as the multiparticle Schrödinger equation in which antisymmetry is essential (e.g., [11, 13, 80, 81, 82]). Imposing structural constraints such as orthogonality on G can simplify the problem (2), but can also produce exponentially inefficient results (e.g., needing $r = 2^d$ instead of $r = 2$ [10, (2.7)]), so we do not consider such constraints or algorithms that depend upon them (e.g., [19, 30, 31, 47, 48, 59, 67, 105, 106, 132]).

There has been steady, but slow, progress in understanding aspects of the approximation problem (2) such as uniqueness (e.g., [26, 35, 36, 71, 93, 94, 110, 111, 112, 116, 118, 119, 120, 121]), maximal and generic rank (e.g., [17, 27, 42]), limits (e.g., [18, 32, 69, 107, 108, 109, 113, 114, 115, 117, 127]), and approximability of function classes (e.g., [19, 24, 47, 48, 50, 51, 133, 134, 135, 136]). However, our understanding in the true tensor case of $d > 2$ is far more primitive than our understanding in the matrix case $d = 2$. We will assume throughout this work that $d > 2$.

It is NP-hard to find G with given r that minimizes the error in (2), to find the minimum r such that there exists G with error less than a given bound, or to solve most other tensor problems [54]. Optimization-based algorithms, which iteratively modify an initial G to reduce the error while keeping r fixed, have been developed based on alternating least squares (ALS) (e.g., [9, 10, 22, 26, 57, 70, 76, 77, 78, 83, 84, 92, 100, 101, 123, 124, 128, 129]), Newton's method (e.g., [38, 62, 95, 125]), nonlinear conjugate gradient (e.g., [2, 34, 39, 91]), line search (e.g., [26, 98]), and others (e.g., [33, 59, 91, 104]); see the discussion and comparisons in [40, 57, 104, 126]. In many cases these algorithms can quickly produce a good quality G. However, in other cases they fail. These failures are a significant impediment to the use of methods based on (2). Indeed, the recent development of alternative tensor formats has largely been motivated by the lack of a satisfactory algorithm to solve (2).

Our ultimate goal is to produce better algorithms for solving (2) so that it can be used more effectively and more widely. Given the unsatisfactory state of affairs after almost 50 years of algorithm development, we believe that a better understanding of the approximation problem itself is needed before such algorithms can be constructed. Moreover, this understanding must focus on aspects that directly affect the algorithms. Our strategy for developing this understanding is to consider (2) as a dynamical system in the parameters defining $\{G_i^l\}$, which flow according to the gradient of an error function. Aspects of this flow, such as the nature of its saddle points, both aid in understanding (2) and are important for optimization algorithms.

We focus our analysis of the flow on understanding the phenomena known informally as "swamps." In its original usage in $d = 3$ (e.g., [26, 79]), a swamp is when $\{G_i^l\}_{l=1}^r$ is nearly linearly dependent for all (three) i's, which has been observed to result in slow convergence of algorithms. We use it more generally (as does [78, 84, 104]) to mean periods of slow convergence, and further distinguish two types of swamps.

- A *terminal swamp* is when the local convergence to a (local) minimum is slow, as illustrated in Figure 1. Such behavior occurs in many optimization problems, and is generally associated with a large condition number in the Hessian of the error function at the minimum.
- A *transient swamp* is when the error decreases by minuscule amounts for many iterations, but then converges rapidly, as illustrated in Figure 2. We know of no other context in which transient swamps naturally occur.

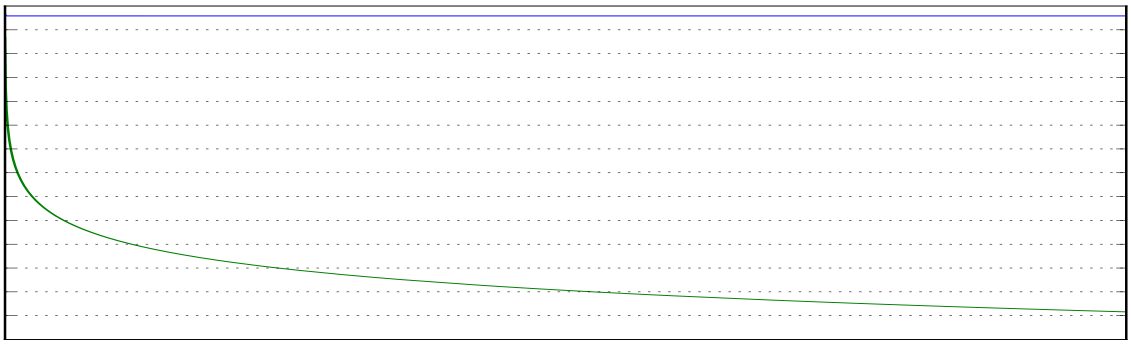

Figure 1. *Illustration of a terminal swamp. The top (blue) curve is* $\log_{10}$ *of the error and the bottom (green) curve is* $\log_{10}$ *of the difference in error of consecutive iterations, with vertical axis* $[-14, 0]$*. The horizontal axis is the iteration number, in* $[0, 10000]$*.*

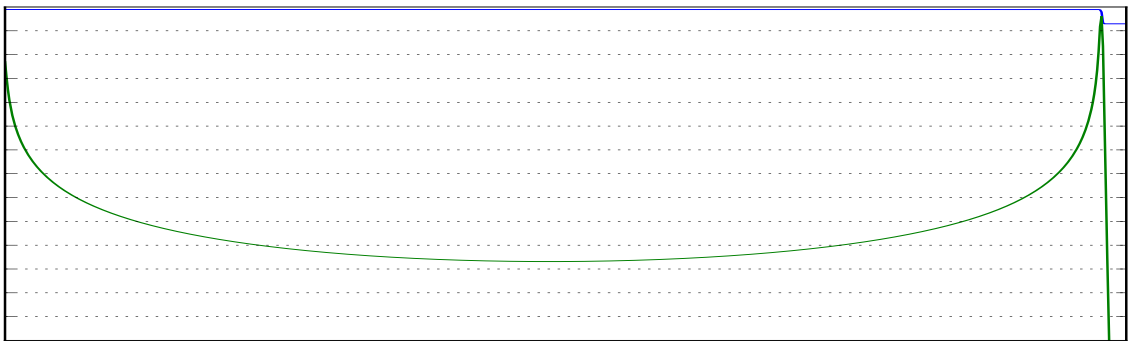

Figure 2. *Illustration of a transient swamp, in the same format as Figure 1, with the iteration number in* $[0, 770]$*.*

The existence of swamps when using the ALS algorithm is well documented and work has been done trying to understand and alleviate them (e.g., [26, 77, 78, 79, 84, 92, 98, 100]). Conditions that correlate with the occurrence of swamps have been identified (i.e., ill-conditioning), but no causal mechanism has been identified. The existence and severity of swamps when using non-ALS algorithms is unresolved. Since the mechanism causing swamps is unknown, there is no way to show theoretically that an algorithm is not affected by it. Numerical tests show that non-ALS algorithms have potential, but do not settle the issue. We want to know the extent to which swamps are caused by high-order local minima, large regions with small gradient away from the minima, slow passage near saddle points, or other phenomena. This knowledge will then allow one to evaluate the extent to which the mechanisms that cause swamps affect various algorithms and identify what features new optimization algorithms should have.

Analysis of (2) as a dynamical system with general T and G is out of reach, so we make simplifying assumptions and consider a model problem.

1. We only consider real tensors and sum-of-separable formats over the reals.
2. We only consider the approximation (2) in the least-squares sense, measuring the error by $\|T - G\|^2$ using the Frobenius norm. We allow an optional additional regularization

term $\lambda \sum_{l=1}^{r} \| \bigotimes_{i=1}^{d} G_i^l \|^2$, which still permits the approximation to be framed as a least-squares problem [83].

3. We only consider T with the lowest possible rank without the problem becoming trivial. If we made T rank-1 (i.e., separable), then, as a consequence of its projection property (see e.g., [83, section 4.1]), a single pass of ALS will reduce the error to zero. Consequently we consider rank-2 T, which can be written as $T = \sum_{l=1}^{2} \bigotimes_{i=1}^{d} T_i^l$.
4. For G we consider only G_1 with too small rank $r = 1$, G_2 with the correct rank $r = 2$, and G_3 with excess rank $r = 3$.

In this paper we develop the framework for analyzing the approximation of rank-2 T by G_1, G_2, and G_3, and conduct the analysis for G_1. The analysis using G_2 and G_3 is in progress, but the analysis using G_1 is already rich and the combined analysis is too long to present at once. The approximation problem (2) with $r = 1$ is distinctly easier than when $r > 1$, since when $r = 1$ a minimizer always exists. Consequently, more is known theoretically about it (e.g., [31, 66, 129, 131, 137]) and some "greedy" algorithms are based on repeatedly fitting with $r = 1$ (e.g., [3, 12, 20, 23, 28, 85, 86, 97]).

The above assumptions are fundamental, defining which problems we consider. We make two additional assumptions, which we characterize as modeling assumptions since they simplify the problem by removing less important parameters. Both assumptions are inspired by the ALS algorithm, which accomplishes them automatically due to its action as an alternating orthogonal projection [83, section 4.1].

5. We track only the directions of the vectors G_i^l, by declaring their norms as fast variables and eliminating those scalars from the analysis. By a simple normalization convection, one can rewrite $G = \sum_{l=1}^{r} a_l \bigotimes_{i=1}^{d} G_i^l$ and assume each G_i^l has norm one. Given $\{G_i^l\}_{i,l}$, the values $\{a_l\}_l$ that minimize $\|T - G\|^2$ can be determined by linear least squares; we assume $\{a_l\}_l$ always maintain these optimal values.
6. We only consider G that satisfy $G_i^l \in \operatorname{span}\{T_i^1, T_i^2\}$, thus removing a portion of G orthogonal to T from the analysis.

Neither the gradient flow nor algorithms other than ALS enforce these assumptions, but they do push the approximation toward these assumptions, since both assumptions reduce the least-squares error.

Given any rank-2 tensor $T = \sum_{l=1}^{2} \bigotimes_{i=1}^{d} T_i^l$, there exist unitary matrixes U_i such that $U_i T_i^1$ is a multiple of the first coordinate basis vector $\mathbf{e}_1 = [1, 0, \dots]$ and $U_i T_i^2$ is in the span of the first two coordinate basis vectors $\{\mathbf{e}_1, \mathbf{e}_2\}$. Applying the separable unitary transformation $\bigotimes_{i=1}^{d} U_i$ to both T and G in (2) simply transforms the approximation problem into a different coordinate system with $T_i^l \mapsto U_i T_i^l$ and $G_i^l \mapsto U_i G_i^l$. Since this transformation is unitary, inner products and the least-squares error are unchanged, and the gradient and Hessian of the least-squares error are the same objects, just in the new coordinate system. Thus the gradient flow commutes with this transformation, and its dynamics are unaffected. Similarly, algorithms based on the inner product, gradient, and Hessian (e.g., gradient descent, conjugate gradient, Newton's method, etc.) commute with this transformation, so their dynamics are unchanged. Since this transformation is also separable and unitary in each direction, ALS also commutes with it and has unchanged dynamics. Although not all algorithms commute with this transformation, the most important ones do, so we allow ourselves to apply it. We

can then assume, without loss of generality, that T_i^1 is a multiple of $\mathbf{e}_1$ and $T_i^2 \in \text{span}\{\mathbf{e}_1, \mathbf{e}_2\}$. Additionally assuming $\|T\| = 1$ and $\|\bigotimes_{i=1}^d T_i^1\| \geq \|\bigotimes_{i=1}^d T_i^2\|$, possibly multiplying by (-1) overall and the unitary matrix $\left[\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right]$ in some directions, and moving scalars through the tensor products, we can put T in the standard form

$$T = \left(1 + 2z \prod_{i=1}^{d} \cos(\phi_i) + z^2\right)^{-1/2} \left(\bigotimes_{i=1}^{d} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + z \bigotimes_{i=1}^{d} \begin{bmatrix} \cos(\phi_i) \\ \sin(\phi_i) \end{bmatrix}\right), \tag{3}$$

where $|z| \leq 1$ and $\phi_i \in [0, \pi/2]$. We thus consider a family of target tensors parameterized by d, z, and $\{\phi_i\}_{i=1}^d$.

By our assumption that $G_i^l \in \text{span}\{T_i^1, T_i^2\}$, when the separable unitary transformation is applied to G, it results in $U_i G_i^l \in \text{span}\{\mathbf{e}_1, \mathbf{e}_2\}$. The nominal vector lengths M_i in (1) then play no role in our analysis. Thus, in the coordinate system induced by (3), we can write

$$G_1 = a \bigotimes_{i=1}^{d} \begin{bmatrix} \cos(\alpha_i) \\ \sin(\alpha_i) \end{bmatrix} \tag{4}$$

and similar expressions for G_2 and G_3. Since d is determined by T and a will be treated as a fast variable, the error will be a function of $\{\alpha_i\}_{i=1}^d$.

In section 2 we develop a framework to analyze minimization problems using dynamical systems concepts. The quantities we consider are related to transient dynamics rather than asymptotic dynamics and so are not part of the usual terminology of dynamical systems.

- We develop a method for measuring the stability of the gradient flow transverse to an invariant set. This allows one to determine if the flow near an invariant (e.g., symmetric) set will converge to or diverge from it and, thus, whether or not analysis on the invariant set captures the dynamics.
- We characterize the behavior of simple gradient-based algorithms in the vicinity of stationary points of the flow and at general points at which the flow is stable in the transverse direction.
- We develop an estimate of the time for the gradient flow, or a simple gradient-based algorithm, from a point to reach a solution. This estimate provides a quantitative way to compare points and models the information that might be available to an approximation algorithm.

The ingredients for these estimates and methods are all well known. Although we have not found them combined in quite these ways in the literature, we assume that they have appeared before.

In section 3 we set up for the analysis of fitting T in (3) by G_1 in (4).

- We show how to easily optimize the scalar coefficient a in G_1, given the angular variables $\{\alpha_i\}$. We then consider a as a "fast" variable, thus reducing the analysis to the angular variables.
- We derive the formulas for the gradient and Hessian of the error with respect to the angular variables.

In section 4 we analyze the case when both T and G_1 are symmetric in direction, meaning $\phi_i = \phi$ and $\alpha_i = \alpha$ for all i.

- We determine when the gradient flow of $\{\alpha_i\}$ along the symmetric set $\alpha_i = \alpha$ is stable with respect to transverse perturbations.
- Under the additional assumption $z = \pm 1$, we analyze the behavior at the symmetry point $\alpha = \phi/2$. For $z = 1$ there is a pitchfork bifurcation with $\alpha = \phi/2$ stable for small ϕ and unstable for larger ϕ.
- We show visualizations for various d's and z's illustrating the dependence on ϕ and α of the error itself, the expected flow time, the expected algorithm time, and the transverse stability. These allow further qualitative description of the dynamics and identification of parameter configurations that would likely cause difficulty for approximation algorithms. In particular, we find:
  - When $z = 1$ and ϕ is slightly less than the bifurcation value, the gradient is very small on a relatively large region, leading to slow convergence of the gradient flow. Moreover, the eigenvalues of the Hessian at the minimum point also indicate slow convergence for algorithms. Thus there is a "terminal" swamp.
  - When $0 < z < 1$, the pitchfork bifurcation becomes a transcritical bifurcation. When ϕ is slightly below the bifurcation value, there is a region of very small gradient at some distance from the minimum point. If an algorithm starts on the wrong side of this region, then it must pass through it and there may be a "transient" swamp.
- We show by numerical experiments that the candidate swamps we identified do indeed result in swamps, which are illustrated in Figures 1 and 2.

In section 5 we analyze the case when T is symmetric but G_1 is not. This section is based in part on the dissertation [45, Chapter 4].

- We describe all global maxima.
- We determine when nonsymmetric stationary points can exist and show that they must have a specific form, with $\{\alpha_i\}$ taking only two values, assuming the stationary point is not a global maximum.
- We analyze the Hessian at these stationary points and thereby show that they are either global maxima or saddles. Thus the gradient flow will eventually lead to a symmetric state.
- We analyze the size of the eigenvalues at these saddles and how they depend on the parameters. We find arbitrarily bad saddles, which are related to a symmetric, nonhyperbolic stationary point with a single positive eigenvalue and the remaining eigenvalues zero. Such bad saddles are more prevalent at smaller angles.

In section 6 we briefly consider the partially symmetric case, where $\{\phi_i\}$ takes on only two values and $\{\alpha_i\}$ shares this partial symmetry. We show that, as expected, bifurcation phenomena with nonhyperbolic stationary points still occur.

We present our analysis in detail in the hopes that the reader will be able to connect it with their own knowledge and thereby advance the understanding of the community. Given our goal of developing understanding of a problem through simple examples, we cannot make any theorem-like conclusions. Instead we summarize the most important points that we now (think that we) understand, and their implications.

1. As the parameters change, the approximation problem undergoes bifurcations, which in some cases lead to qualitative changes in the nature of the problem. It is likely that

different algorithms are better in the different regimes, and thus a good method should be able to switch algorithms. This effect may also explain why proposed algorithms look better in the articles that proposed them, where the authors chose the test cases, than in comparison studies [40, 57, 126].

2. The worst features, in the form of nonhyperbolic stationary points, occur at discrete, nonzero angles. In the literature, swamps are associated with small angles (called ill-conditioning or degeneracy) (see e.g., [26, 78, 79, 84, 98, 100]), thereby suggesting that the worst case should be the limit as the angle goes to zero. Our results are compatible with observations in the literature if "small" angle is interpreted as being on the small side of a bifurcation rather than as a limit.
3. Transverse stability slows algorithms by causing them to make slower progress in the flow direction. Essentially, gradient-based algorithms must take small steps because as they begin to run up the opposite side of a valley they stop. It has been observed that when in a swamp the ALS algorithm makes many small steps in the same general direction (e.g., [16, 52, 57, 76, 98, 126]). We believe that our analysis using transverse stability explains such behavior. This analysis supports the use of extrapolation methods (e.g., [16, 33, 52, 57, 76, 98, 126]) and methods based on nonlinear conjugate gradient (e.g., [2, 34, 39, 91]).

2. Analysis framework.

In this section we develop a framework for analyzing an approximation problem using dynamical systems concepts. In subsection 2.1 we review the dynamical systems background, such as the gradient flow and how it behaves near stationary points. In subsection 2.2 we develop a method for measuring the stability of the gradient flow transverse to an invariant set. In subsection 2.3 we characterize the behavior of simple gradient-based algorithms in the vicinity of features of the gradient flow. In subsection 2.4 we develop an estimate of the time for the gradient flow, or a gradient-based algorithm, from a point to reach a solution.

Some of these methods are nonstandard and we have been unable to locate them in the literature. On the other hand, they are all based on combinations of the objective function value, the norm of its gradient, and the eigenvalues of its Hessian, so it would be surprising if they were truly new.

We let $\mathbf{1}_k$ be the (column) vector of size k with entries 1 and I_k be the identity matrix of size $k \times k$; if the subscript k is missing it is assumed to be d. Let $\mathbf{e}_j$ be the vector with 1 in entry j and otherwise 0, $\mathbf{x}^*$ be the (Hermitian) transpose, and $\|\cdot\|$ be the L^2 (Frobenius) norm. For $1 \le q \le p$, let the matrix

$$U_{pq} = I_p - 2\frac{(\mathbf{e}_q - \mathbf{1}_p/\sqrt{p})(\mathbf{e}_q - \mathbf{1}_p/\sqrt{p})^*}{(\mathbf{e}_q - \mathbf{1}_p/\sqrt{p})^*(\mathbf{e}_q - \mathbf{1}_p/\sqrt{p})} \tag{5}$$

be the Householder reflector [58] that takes $\mathbf{1}_p$ to $\sqrt{p}\mathbf{e}_q$; it is unitary and is its own conjugate transpose and inverse.

2.1. Dynamical systems background.

Suppose we have a differentiable function $f(\mathbf{x})$ defined on some connected domain D. By the *gradient flow* of f we mean the flow induced by the differential equation

$$\dot{\mathbf{x}}(t) = -\nabla f(\mathbf{x}(t)). \tag{6}$$

We call f the *objective function* of the flow. We use the convention of the negative gradient in (6) because we will ultimately wish to minimize the error of the approximation (2), as measured by some norm of the difference between T and G. It is a basic fact about gradient flows in general that any solution of (6) that is bounded (remains inside a compact subset of D for all $t \geq 0$) will converge to some subset of the set of critical points of f [55]. Since we have assumed that f is differentiable, the critical points of f are the points x where $\nabla f(\mathbf{x}) = 0$, which are precisely the *equilibirum* or *stationary* points of the flow defined by (6).

The stability of the flow at an equilibrium $\mathbf{x}$ is often a starting point for understanding a dynamical system. It is well known that the local stability at $\mathbf{x}$ can be inferred in many cases by the eigenvalues of the Jacobian of the vector field at the point. In the case of a gradient flow (6) the Jacobian of the vector field is exactly the Hessian matrix of the objective function $f(\mathbf{x})$. Since the Hessian is symmetric, it has only real eigenvalues. Without loss of generality, suppose f has a critical point at $\mathbf{0}$ and $f(\mathbf{0}) = 0$. Letting H denote the Hessian of f at $\mathbf{0}$, we have

$$f(\mathbf{x}) \approx \frac{1}{2}\mathbf{x}^* H\mathbf{x} \qquad \text{and} \qquad \nabla f(\mathbf{x}) \approx H\mathbf{x}\,.$$

Let μ be the smallest eigenvalue of H. We will only discuss the *hyperbolic* case, when zero is not an eigenvalue of H.

- If all the eigenvalues of H are positive, then $\mathbf{0}$ is a local minimum. In particular, $\mathbf{0}$ is locally exponentially stable, meaning that all points in some small neighborhood of $\mathbf{0}$ remain in a neighborhood of the point and converge to $\mathbf{0}$ with an exponential rate. Asymptotically, for a generic $\mathbf{x}$ near $\mathbf{0}$ the distance changes like $\exp(-\mu t)$.
- If all the eigenvalues of H are negative, then the stationary point is a local maximum, and $\mathbf{0}$ is locally exponentially unstable. All $\mathbf{x} \neq \mathbf{0}$ near $\mathbf{0}$ will flow away from $\mathbf{0}$ with distance changing like $\exp(-\mu t)$; note that now $\mu < 0$.
- If H has both positive and negative eigenvalues, then $\mathbf{0}$ is a saddle point. Generic $\mathbf{x}$ near $\mathbf{0}$ flow near $\mathbf{0}$ for some time but eventually flow away from it with distance changing like $\exp(-\mu t)$; note that $\mu < 0$.

Away from stationary points, the flow moves like $t\|\nabla f(\mathbf{x})\|$, and the Hessian plays no role.

2.2. Measuring stability transverse to the flow on an invariant set. In this section we describe a general procedure for measuring the stability of the trajectories in a gradient flow. We will use it to determine when we can restrict our analysis to a smaller (e.g., symmetric) set and when we need to consider the whole parameter space. Since this method is rather simple, we expect that it is known, but we have not found it in the literature.

Consider again a differentiable function $f(\mathbf{x})$ with parameters evolving under the gradient flow defined by $\dot{\mathbf{x}}(t) = -\nabla f(\mathbf{x}(t))$. We can consider two stability questions about the flow:

1. Given some value $\mathbf{x}$ and a nearby value $\mathbf{x}_1$, will the flows from $\mathbf{x}$ and $\mathbf{x}_1$ locally converge or diverge?
2. Given that the flow from $\mathbf{x}$ should stay in some invariant set, due, for example, to symmetry, if we start from a nearby value $\mathbf{x}_1$ not in this set, will the flow from $\mathbf{x}_1$ converge to or diverge from this set?

In both cases the strategy is the following.

1. Compute the Hessian at $\mathbf{x}$, denoted $H(\mathbf{x})$.

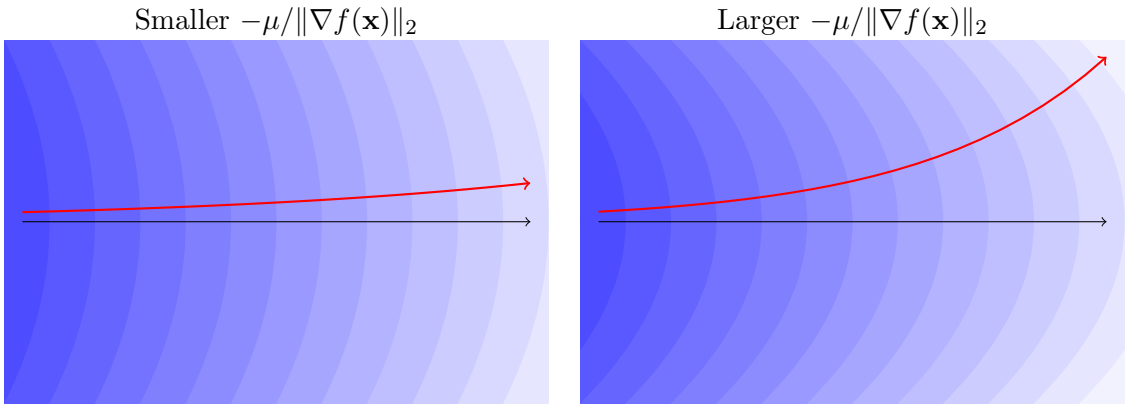

Figure 3. *Illustration of the geometric divergence from the negative gradient direction for $\mu < 0$. The negative gradient points to the right and the flow moves away from it an amount determined by the relative sizes of $-\mu$ and $\|\nabla f(\mathbf{x})\|_2$.*

2. Apply a change of coordinates so that either $\nabla f(\mathbf{x})$ or an orthogonal basis for the invariant set are in some specific coordinate directions.
3. Delete those coordinates to obtain the "transverse" Hessian, denoted $H^{\perp}(\mathbf{x})$.
4. Compute the smallest eigenvalue of $H^{\perp}(\mathbf{x})$ and denote it as $\mu(\mathbf{x})$. If $\mu(\mathbf{x}) > 0$, then the flow is stable, if $\mu(\mathbf{x}) < 0$ the flow is unstable, and if $\mu(\mathbf{x}) = 0$, then the flow is (linearly) neutral.

The flow moves in the negative gradient direction like $x = \|\nabla f(\mathbf{x})\|_2 t$ and toward or away from the unperturbed flow like $y = \exp(-\mu(\mathbf{x})t)$. Substituting to eliminate t yields the geometric behavior $y = \exp(-h(\mathbf{x})x)$ with

$$h(\mathbf{x}) = \frac{\mu(\mathbf{x})}{\|\nabla f(\mathbf{x})\|_2} \quad \in [-\infty, \infty]. \tag{7}$$

For an illustration of this effect, see Figure 3. Although the sign of $\mu(\mathbf{x})$ is sufficient to determine *if* the flow is (un)stable at $\mathbf{x}$, the magnitude of $\mu(\mathbf{x})$ does not allow meaningful comparisons of *how* (un)stable the flow is at two different points. To allow such comparisons we will use $h(\mathbf{x})$.

2.3. Algorithm progress rate. The behavior of the flow described in subsection 2.1 gives some information about the behavior of gradient-based algorithms, but can also be misleading. In this section we discuss the behavior of the simplest gradient-based algorithm near stationary points and in a general position. We consider the gradient-descent minimization algorithm with line search: From an initial point $\mathbf{x}$, compute the gradient $\nabla f(\mathbf{x})$, find t to minimize $f(\mathbf{x} - t\nabla f(\mathbf{x}))$, set $\mathbf{x}_{\text{new}} = \mathbf{x} - t\nabla f(\mathbf{x})$, and then repeat. Gradient descent is generally considered a poor algorithm, but it serves well to illustrate the differences between the flow and the behavior of an algorithm.

Recall that the gradient flow near a stationary point is like $\exp(-\mu t)$, where μ is the smallest eigenvalue of the Hessian. We show the following:

- At a maximum, the algorithm escapes in one step, so the size of the eigenvalues of H is irrelevant, as long as they are all negative.
- At a minimum, the progress of the algorithm is governed not by μ, but by the ratio μ/η, where η is the maximum eigenvalue of H.
- At a saddle, the progress is also governed by the ratio μ/η.

The behavior of gradient descent at a minimum is well known (see e.g., [96, 103]), and is a prime motivation for using conjugate-gradient algorithms (see e.g., [41, 44, 96]). We have not found the corresponding analysis for saddles in the literature, but we assume that it is also known.

Away from stationary points, the gradient flow moves like $\|\nabla f(\mathbf{x})\|$. Under the assumption that the minimum eigenvalue of the transverse Hessian from subsection 2.2 is positive, we find that the progress of the algorithm is governed by the ratio $\|\nabla f(\mathbf{x})\|/\eta$, where η is the maximum eigenvalue of the transverse Hessian. We have not found this analysis in the literature, but we expect that it is known; certainly it is known that algorithms need to be able to handle such features (called valleys in, e.g., [41] and ridges in [52, p. 31]).

2.3.1. Algorithm progress near stationary points. As in subsection 2.1, to analyze stationary points we take $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^* H\mathbf{x}$ and so $\nabla f(\mathbf{x}) = H\mathbf{x}$. Along the negative gradient from a starting point $\mathbf{x}$, the objective function is

$$f(\mathbf{x} - t\nabla f(\mathbf{x})) = \frac{1}{2}(\mathbf{x} - tH\mathbf{x})^* H(\mathbf{x} - tH\mathbf{x}) = \frac{t^2}{2}\mathbf{x}^* H^3\mathbf{x} - t\mathbf{x}^* H^2\mathbf{x} + \mathbf{x}^* H\mathbf{x}\,. \tag{8}$$

If $\mathbf{x}^* H^3\mathbf{x} < 0$, then (8) has no minimum with respect to t, so the gradient-descent minimization algorithm would move to infinity. If $\mathbf{x}^* H^3\mathbf{x} > 0$, then the minimum of (8) occurs when

$$t = \frac{\mathbf{x}^* H^2\mathbf{x}}{\mathbf{x}^* H^3\mathbf{x}}\,, \quad \text{which yields} \qquad\qquad \mathbf{x}_{\text{new}} = \mathbf{x} - \frac{\mathbf{x}^* H^2\mathbf{x}}{\mathbf{x}^* H^3\mathbf{x}} H\mathbf{x}\,.$$

At a maximum all eigenvalues of H are negative, so $\mathbf{x}^* H^3\mathbf{x} < 0$ and the algorithm moves far from the stationary point in one step and the flow rate in subsection 2.1 is irrelevant. At a minimum $\mathbf{x}^* H^3\mathbf{x} > 0$ so the algorithm will take a finite step toward the stationary point. At a saddle the sign of $\mathbf{x}^* H^3\mathbf{x}$ depends on $\mathbf{x}$.

To understand the minimum and saddle cases, we consider the simplest example, where H is 2×2 and diagonal, with the two eigenvalues $\mu < \eta$ with $0 < \eta$. Thus we have

$$H = \begin{bmatrix} \mu & 0 \\ 0 & \eta \end{bmatrix}\,, \quad \text{which yields}$$

$$\mathbf{x}_{\text{new}} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - \frac{\mu^2 x_1^2 + \eta^2 x_2^2}{\mu^3 x_1^2 + \eta^3 x_2^2} \begin{bmatrix} \mu x_1 \\ \eta x_2 \end{bmatrix} = \frac{\left(1 - \frac{\mu}{\eta}\right)}{1 + \left(\frac{\mu}{\eta}\right)^3 \left(\frac{x_1}{x_2}\right)^2} \begin{bmatrix} x_1 \\ -\left(\frac{\mu}{\eta}\right)^2 \left(\frac{x_1}{x_2}\right)^2 x_2 \end{bmatrix}\,.$$

The second update is

$$(\mathbf{x}_{\text{new}})_{\text{new}} = \frac{\left(1 - \frac{\mu}{\eta}\right)^2}{1 + \frac{\mu}{\eta}\left(\left(\frac{\mu}{\eta}\frac{x_1}{x_2}\right)^2 + \left(\frac{\mu}{\eta}\frac{x_1}{x_2}\right)^{-2}\right) + \left(\frac{\mu}{\eta}\right)^2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\,, \tag{9}$$

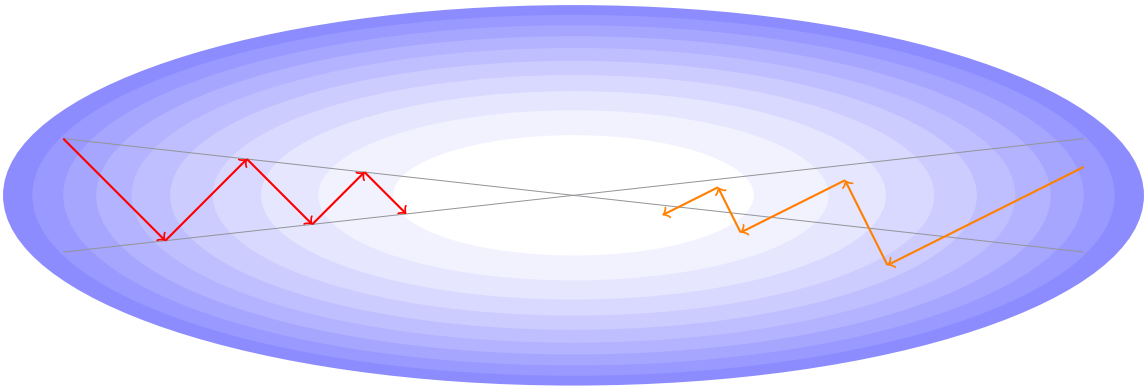

Figure 4. *Convergence behavior of gradient descent with line search near a local minimum. The red path on the left is slowest since it started on a line of slope* $\pm\mu/\eta$.

which is on the same ray as $\mathbf{x}$. As we continue to update, every two updates multiplies by the scalar in (9).

For a minimum, we have $0 < \mu < \eta$ and the scalar in (9) is in $(0, 1)$, with smaller values giving faster convergence toward the minimum $\mathbf{x} = \left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$. With $0 < \mu < \eta$ fixed, the scalar is maximized when $|(\mu x_1)/(\eta x_2)| = 1$, which then gives the worst-case (i.e., slowest) convergence. Taking the square root to give the net contraction factor per iteration then gives the scalar

$$\frac{1-\mu/\eta}{1+\mu/\eta}. \tag{10}$$

Thus convergence is fastest as $\mu/\eta \to 1^-$ and slowest as $\mu/\eta \to 0^+$. See Figure 4 for an illustration.

For a saddle we have $\mu < 0 < \eta$ and there are three possible behaviors. First, if $\mathbf{x}^* H^3 \mathbf{x} < 0$, then the algorithm moves far from the stationary point in a single step. Second, if $\mathbf{x}^* H^3 \mathbf{x} > 0$ but $\mathbf{x}_{\text{new}}^* H^3 \mathbf{x}_{\text{new}} < 0$, then the algorithm moves far from the stationary point in the second step. Third, if $\mathbf{x}^* H^3 \mathbf{x} > 0$ and $\mathbf{x}_{\text{new}}^* H^3 \mathbf{x}_{\text{new}} > 0$, then the iteration proceeds according to (9). In terms of ratios, these conditions are

$$\mathbf{x}^* H^3 \mathbf{x} > 0 \Leftrightarrow \mu^3 x_1^2 + \eta^3 x_2^2 > 0 \Leftrightarrow \left|\frac{x_1}{x_2}\right| < \left(\frac{-\mu}{\eta}\right)^{-3/2} \quad \text{and}$$

$$\mathbf{x}_{\text{new}}^* H^3 \mathbf{x}_{\text{new}} > 0 \Leftrightarrow \left| -\left(\frac{\mu}{\eta}\right)^{-2} \left(\frac{x_1}{x_2}\right)^{-1} \right| < \left(\frac{-\mu}{\eta}\right)^{-3/2} \Leftrightarrow \left(\frac{-\mu}{\eta}\right)^{-1/2} < \left|\frac{x_1}{x_2}\right|,$$

which can be combined into

$$\left|\frac{x_1}{x_2}\right| \in \left(\left(\frac{-\mu}{\eta}\right)^{-1/2}, \left(\frac{-\mu}{\eta}\right)^{-3/2} \right). \tag{11}$$

Note that if $\mu/\eta \le -1$, then this interval is empty so the algorithm will move far from the stationary point in one or two steps regardless of the starting position $\mathbf{x}$. Assuming (11) holds,

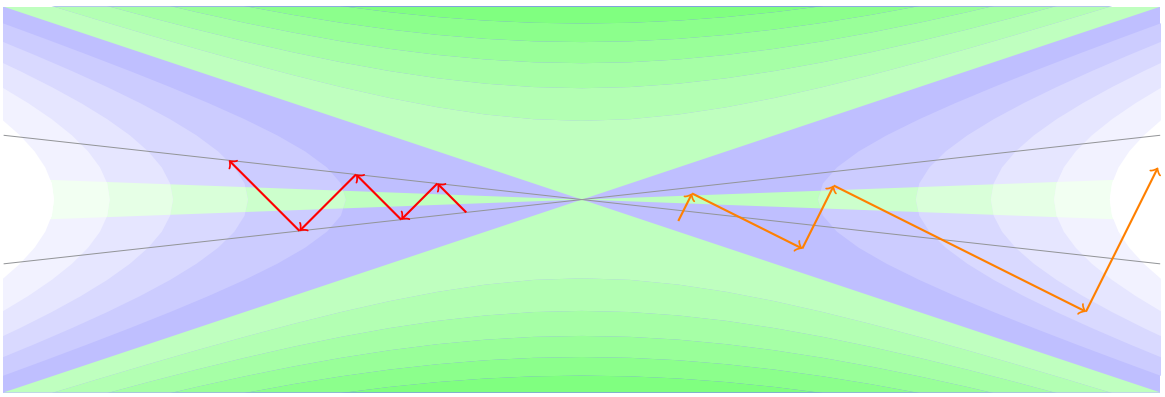

Figure 5. *Divergence behavior of gradient descent with line search near a saddle point. The red path on the left is slowest since it started on a line of slope $\pm\mu/\eta$. In the green regions the iteration escapes in one or two steps.*

the scalar in (9) is in $(1,\infty)$, with larger values giving faster movement away from the saddle. With $\mu < 0 < \eta$ fixed, the scalar is minimized when $|(\mu x_1)/(\eta x_2)| = 1$ and yields the net contraction factor per iteration of (10); notice that now $\mu/\eta < 0$. Thus divergence is fastest as $\mu/\eta \to -1^+$ and slowest as $\mu/\eta \to 0^-$. See Figure 5 for an illustration.

In subsection 2.1, the flow moved toward the minimum or away from a saddle like $\exp(-\mu t)$. Here we found the algorithm moves like $(10)^k$ in k steps. To understand the difference, we can either compare $\exp(-\mu)$ to (10) or compare μ to

$$-\ln((10)) = -\ln\left(\frac{1-\frac{\mu}{\eta}}{1+\frac{\mu}{\eta}}\right) = \ln\left(1+\frac{\mu}{\eta}\right) - \ln\left(1-\frac{\mu}{\eta}\right) = \frac{2\mu}{\eta} + \mathcal{O}\left(\left(\frac{\mu}{\eta}\right)^3\right). \tag{12}$$

For more than two eigenvalues the iteration is not clean like (9). We will let μ be the minimum eigenvalue of H and η be the maximum eigenvalue. For a minimum, the ratio η/μ is the condition number of H and the contraction factor (10) has been established as a bound (see e.g., [103] which uses [61]). For a saddle, μ/η is the ratio of the strongest repelling force to the strongest attracting force; we believe this is the correct quantity to consider, but do not have rigorous justification. We will use μ/η to compare saddles and for plotting; in both cases we only use it when $\mu/\eta \in [-1,0]$ since $\mu/\eta < -1$ leads to escape in one or two steps.

2.3.2. Algorithm progress far from stationary points. To understand the flow far from stationary points, we consider an objective function centered at $\mathbf{x}$ with the negative gradient of norm δ pointing in the $\mathbf{e}_1$ direction and diagonal Hessian with eigenvalue 0 corresponding to $\mathbf{e}_1$ and eigenvalue η corresponding to $\mathbf{e}_2$. (Thus η is the eigenvalue of the transverse Hessian.) Thus we have

$$f(\mathbf{x}) = \mathbf{x}^* \begin{bmatrix} -\delta \\ 0 \end{bmatrix} + \frac{1}{2}\mathbf{x}^* \begin{bmatrix} 0 & 0 \\ 0 & \eta \end{bmatrix} \mathbf{x} = -\delta x_1 + \frac{1}{2}\eta x_2^2 . \tag{13}$$

The gradient and Hessian of f are

$$\nabla f(\mathbf{x}) = \begin{bmatrix} -\delta \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \eta \end{bmatrix} \mathbf{x} = \begin{bmatrix} -\delta \\ \eta x_2 \end{bmatrix} \quad \text{and} \quad H(\mathbf{x}) = \begin{bmatrix} 0 & 0 \\ 0 & \eta \end{bmatrix}.$$

We are interested in the progress of the algorithm in the $\mathbf{e}_1$ direction, which means the change in x_1.

Along the negative gradient from a starting point, the objective function is

$$f(\mathbf{x} - t\nabla f(\mathbf{x})) = f\left(\begin{bmatrix} x_1 + t\delta \\ x_2 - t\eta x_2 \end{bmatrix}\right) = \frac{t^2}{2}\eta^3 x_2^2 - t(\delta^2 + \eta^2 x_2^2) - \delta x_1 + \frac{1}{2}\eta x_2^2. \tag{14}$$

If $\eta < 0$, then (14) has no minimum, so the gradient-descent minimization algorithm would move to infinity. If $\eta > 0$ and $x_2 \neq 0$, then the minimum of (14) occurs when

$$t = \frac{\delta^2 + \eta^2 x_2^2}{\eta^3 x_2^2}, \quad \text{which yields}$$

$$\mathbf{x}_{\text{new}} = \mathbf{x} - \frac{\delta^2 + \eta^2 x_2^2}{\eta^3 x_2^2}\begin{bmatrix} -\delta \\ \eta x_2 \end{bmatrix} = \begin{bmatrix} x_1 + \frac{\delta}{\eta}\left(1 + \left(\frac{\delta}{\eta}\right)^2 x_2^{-2}\right) \\ -\left(\frac{\delta}{\eta}\right)^2 x_2^{-1} \end{bmatrix} \quad \text{and}$$

$$(\mathbf{x}_{\text{new}})_{\text{new}} = \begin{bmatrix} x_1 + \frac{\delta}{\eta}\left(2 + \left(\frac{\delta}{\eta x_2}\right)^2 + \left(\frac{\delta}{\eta x_2}\right)^{-2}\right) \\ x_2 \end{bmatrix}. \tag{15}$$

The $\mathbf{e}_2$ entry has period two. Dividing by two to account for two iterations, the $\mathbf{e}_1$ entry changes by

$$\frac{\delta}{\eta}\left(1 + \frac{1}{2}\left(\left(\frac{\delta}{\eta x_2}\right)^2 + \left(\frac{\delta}{\eta x_2}\right)^{-2}\right)\right),$$

which attains its minimum value of $2\delta/\eta$ when $|x_2| = \delta/\eta$. In subsection 2.1, the flow moved like $t\|\nabla f(\mathbf{x})\|$, but here we find worst-case progress like $k2\|\nabla f(\mathbf{x})\|/\eta$. See Figure 6 for an illustration.

If the transverse Hessian has eigenvalues other than η, then the iteration will not be so clean. If all its eigenvalues are positive we can take η to be the maximum eigenvalue. If it has both positive and negative eigenvalues, then we expect behavior like a saddle in the transverse direction with movement along it due to the gradient; we will not attempt to analyze that case any further.

2.4. Converting rates to estimated times. Suppose we have two points $\mathbf{x}_1$ and $\mathbf{x}_2$ with $f(\mathbf{x}_1) < f(\mathbf{x}_2)$. Given only this information, we would prefer the point $\mathbf{x}_1$ to the point $\mathbf{x}_2$. Now suppose we also know $\|\nabla f(\mathbf{x}_1)\| < \|\nabla f(\mathbf{x}_2)\|$, so that the gradient flow from $\mathbf{x}_1$ is slower than the gradient flow from $\mathbf{x}_2$. It is then not clear which point we should prefer. In this section we develop a quantity that can be used to compare points based on both $f(\mathbf{x})$ and $\|\nabla f(\mathbf{x})\|$. This quantity is interpreted as a local estimate of the time it would take for the flow from $\mathbf{x}$ to reach a root (which is also a minimum) of f.

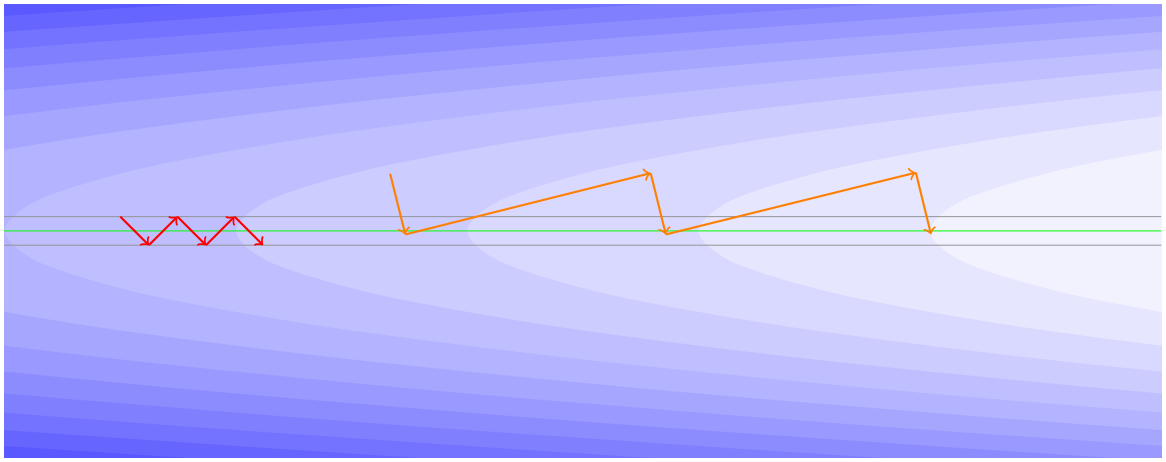

Figure 6. *Behavior of gradient descent with line search in a valley. The red path on the left is slowest since it started on a line with $x_2 = \pm\delta/\eta$.*

Table 1
Examples of estimated flow times.

$f(x)$	$x(t)$	$f(x(t))$	$\lvert f(x)/f'(x)\rvert$	$s(x)$
x^2	$C_2\exp(-2t)$	$C_2^2\exp(-4t)$	$x/2$	$1/4$
x^4	$\pm(C_4+8t)^{-1/2}$	$(C_4+8t)^{-2}$	$x/4$	$(4x)^{-2}$

Suppose we have a differentiable function $f(\mathbf{x})$ defined on some connected domain D such that $\text{range}(f) = \{f(\mathbf{x}) \,|\, \mathbf{x} \in D\} = [0,1]$. We would like to find a root $\mathbf{x}_*$ such that $f(\mathbf{x}_*) = 0$ by following the gradient flow $\dot{\mathbf{x}}(t) = -\nabla f(\mathbf{x}(t))$ from some initial condition $\mathbf{x}$. Since the root $\mathbf{x}_*$ is in general unknown, we cannot measure $\|\mathbf{x} - \mathbf{x}_*\|$. However, we can estimate $\|\mathbf{x} - \mathbf{x}_*\| \approx f(\mathbf{x})/\|\nabla f(\mathbf{x})\|_2$, as is done in Newton's method. Since the speed of the gradient flow is $\|\nabla f(\mathbf{x})\|_2$, the estimated flow time is

$$s(\mathbf{x}) = \frac{f(\mathbf{x})}{\|\nabla f(\mathbf{x})\|_2}\frac{1}{\|\nabla f(\mathbf{x})\|_2} = \frac{f(\mathbf{x})}{\|\nabla f(\mathbf{x})\|_2^2}\,. \tag{16}$$

We will use the quantity $s(\mathbf{x})$ as a way to compare points, with smaller values considered better. For an algorithm, we replace the flow speed $\|\nabla f(\mathbf{x})\|_2$ by the algorithm progress rate $2\|\nabla f(\mathbf{x})\|_2/\eta$, as determined in subsection 2.3.2, to obtain

$$v(\mathbf{x}) = \frac{\eta}{2}s(\mathbf{x}) = \frac{f(\mathbf{x})\eta}{2\|\nabla f(\mathbf{x})\|_2^2} \quad \text{if } 0 < \mu \tag{17}$$

as an estimate of the algorithm time, with no estimate when $\mu < 0$.

For a differentiable f with $\mathbf{x}_*$ in the interior of D, the flow will not reach $\mathbf{x}_*$ in finite time, so the estimate (16) is suspicious. Consider what happens for some simple examples. Table 1 shows what happens for the examples of $f(x) = x^2$ and $f(x) = x^4$ near a root $x_* = 0$.

Although convergence is much faster in the x^2 case, the ratio $|f(x)/f'(x)|$ goes to 0 linearly in both cases and so fails to distinguish them. In contrast, $s(x)$ goes to a finite limit for x^2 and diverges for x^4, thus distinguishing the cases and labeling the x^4 case as worse. So, while it is hard to say exactly what $s(\mathbf{x})$ is measuring near the minimum, it is providing useful qualitative information.

The quantity $s(\mathbf{x})$ is also connected to the convergence theory using the Łojasiewicz gradient inequality. The Łojasiewicz gradient inequality, which holds for real analytic f, states that for all $\mathbf{x}_*$ there exist $C > 0$, $\theta \in (1, 2]$, and an open neighborhood of $\mathbf{x}_*$, such that for all $\mathbf{x}$ in this neighborhood,

$$|f(\mathbf{x}) - f(\mathbf{x}_*)| \leq C\|\nabla f(\mathbf{x})\|_2^{\theta}. \tag{18}$$

Under certain assumptions on an iterative, gradient-based method for minimizing f, its iterates $\mathbf{x}_k$ will approach a local minimizer $\mathbf{x}_*$ and the error will satisfy

$$\|\mathbf{x}_* - \mathbf{x}_k\| = \begin{cases} \mathcal{O}(q^k) & \text{if } \theta = 2 \text{ (for some } 0 < q < 1), \\ \mathcal{O}(k^{-(\theta-1)/(2-\theta)}) & \text{if } 1 < \theta < 2. \end{cases} \tag{19}$$

Thus linear convergence is only established when (18) holds with $\theta = 2$ (for that local minimizer $\mathbf{x}_*$). For general discussion of this theory see [1, 102] and for its use in tensor approximations see [128, 129]. Since we desire to approach the global minimizer, we set $f(\mathbf{x}_*) = 0$. Since we desire to approach with at least linear convergence, we set $\theta = 2$. With these choices, (18) is equivalent to $s(\mathbf{x}) \leq C$ holding uniformly in a neighborhood of $\mathbf{x}_*$. Pointwise, $s(\mathbf{x})$ estimates how well an iterative method starting at $\mathbf{x}$ can be expected to converge to an $\mathbf{x}_*$ with $f(\mathbf{x}_*) = 0$. Small values of $s(\mathbf{x})$ indicate that (18) is easily satisfied and linear convergence is expected. Large values indicate that (18) either requires a large C, which can be expected to slow convergence by increasing q, or that it requires a smaller θ, which would mean slower than linear convergence.

If we remove the assumption that there exists $\mathbf{x}_*$ such that $f(\mathbf{x}_*) = 0$ and have a global minimizer with $f(\mathbf{x}_*) > 0$, then $s(\mathbf{x})$ will diverge at the global minimizer as well. This behavior is not necessarily undesirable, as we can use it to tell that $f(\mathbf{x})$ is unlikely to decrease much further. In the (unlikely) event that we know the value of $f(\mathbf{x}_*)$, we could subtract it from $f(\mathbf{x})$.

3. Fitting a rank-2 tensor with a rank-1 tensor.

In this section we set notation and derive the general formulas for fitting the rank-2 target T in (3) by the rank-1 tensor $G_1 = (4) = a \bigotimes_{i=1}^{d} \begin{bmatrix} \cos(\alpha_i) \\ \sin(\alpha_i) \end{bmatrix}$. Subsection 3.1 establishes notation and other preliminaries. In subsection 3.2 we describe how to make the scalar coefficient a into a fast variable so that the error only depends on the angular variables. In subsection 3.3 we determine how the error, gradient, and Hessian depend on these angular variables.

3.1. Notation and other preliminaries.

For any two (real-valued) tensors F and G indexed by $j_i = 1, 2, \ldots, M_i$ for $i = 1, \ldots d$, define the inner product by

$$\langle F, G \rangle = \sum_{j_1=1}^{M_1} \cdots \sum_{j_d=1}^{M_d} F(j_1, \ldots, j_d) G(j_1, \ldots, j_d)$$

and norm by $\|F\| = \sqrt{\langle F, F\rangle}$. If

$$F = \sum_{m=1}^{R} f_m \bigotimes_{i=1}^{d} F_i^m \quad \text{and} \quad G = \sum_{l=1}^{r} g_l \bigotimes_{i=1}^{d} G_i^l\,,$$
$$\text{then} \quad \langle F, G\rangle = \sum_{m=1}^{R}\sum_{l=1}^{r} f_m g_l \prod_{i=1}^{d} \left\langle F_i^m, G_i^l \right\rangle .$$

We measure the quality of the approximation using regularized least-squares error $\|T-G\|^2 + \lambda\sum_{l=1}^{r}\|\bigotimes_{i=1}^{d} G_i^l\|^2$, which plays the role of f in section 2. For $G_1 = (4) = a\bigotimes_{i=1}^{d}\begin{bmatrix}\cos(\alpha_i)\\ \sin(\alpha_i)\end{bmatrix}$, this error reduces to

$$E_\lambda(G_1) = \|T - G_1\|^2 + \lambda\left(a^2\right) \tag{20}$$

for some $\lambda \geq 0$. When $\lambda = 0$ this reduces to ordinary least-squares error. For G_1, λ does not play an important role; we retain it for later connections with the G_2 and G_3 cases.

We define

$$n(\boldsymbol{\alpha}) = \left\langle \bigotimes_{i=1}^{d}\begin{bmatrix}c\cos(\alpha_i)\\ \sin(\alpha_i)\end{bmatrix}, T\right\rangle = \frac{\prod_{i=1}^{d}\cos(\alpha_i) + z\prod_{i=1}^{d}\cos(\alpha_i - \phi_i)}{\left(1 + 2z\prod_{i=1}^{d}\cos(\phi_i) + z^2\right)^{1/2}}\,, \tag{21}$$

which is the inner product of a normalized rank-1 tensor with the target. Its first partial derivatives are

$$n_j(\boldsymbol{\alpha}) = \frac{\partial}{\partial\alpha_j} n(\boldsymbol{\alpha}) = \frac{-\sin(\alpha_j)\prod_{i=1,\neq j}^{d}\cos(\alpha_i) - z\sin(\alpha_j - \phi_j)\prod_{i=1,\neq j}^{d}\cos(\alpha_i - \phi_i)}{\left(1 + 2z\prod_{i=1}^{d}\cos(\phi_i) + z^2\right)^{1/2}}\,. \tag{22}$$

When $j \neq k$, its second partial derivatives are

$$\begin{aligned} n_{jk}(\boldsymbol{\alpha}) &= \frac{\partial}{\partial\alpha_k} n_j(\boldsymbol{\alpha}) \\ &= \frac{\sin(\alpha_j)\sin(\alpha_k)\prod_{i=1,\neq j,k}^{d}\cos(\alpha_i) + z\sin(\alpha_j - \phi_j)\sin(\alpha_k - \phi_k)\prod_{i=1,\neq j,k}^{d}\cos(\alpha_i - \phi_i)}{\left(1 + 2z\prod_{i=1}^{d}\cos(\phi_i) + z^2\right)^{1/2}} \end{aligned} \tag{23}$$

and when $j = k$ they are

$$n_{jj}(\boldsymbol{\alpha}) = \frac{\partial}{\partial\alpha_j} n_j(\boldsymbol{\alpha}) = -n(\boldsymbol{\alpha})\,. \tag{24}$$

3.2. Making the scalar into a fast variable. Our approximation G_1 is defined by the angles $\{\alpha_i\}_{i=1}^{d}$ and the scalar a. Given the angles, the optimal scalar can be determined by a linear least-squares process. To avoid uninformative complications, we will always assume the scalar is set to its optimal value. Thinking of the approximation problem as a dynamical process in the angular variables, we effectively make the scalar into a fast variable that

instantly optimizes itself. In certain cases the use of fast variables is rigorously justified (see e.g., [53, 60]). We do not use this theory, but instead simply note that algorithms such as ALS can easily maintain optimal scalar coefficients.

In this section we give the formulas/algorithms to determine the optimal scalar value for G_1. We formulate the method in a way that generalizes to the G_2 and G_3 cases. For G_1 this leads to the silly situation of solving a 1×1 linear system.

Once we have fixed $\{\phi_i\}_{i=1}^d$, z, and $\{\alpha_i\}_{i=1}^d$, then we can consider the target as a fixed unit vector and the separable term in G_1 as a unit vector. The minimum of $E_\lambda(G_1)$ is the least-squares solution to the system

$$\begin{bmatrix} \bigotimes_{i=1}^d \begin{bmatrix} \cos(\alpha_i) \\ \sin(\alpha_i) \end{bmatrix} \\ \sqrt{\lambda} \bigotimes_{i=1}^d \begin{bmatrix} \cos(\alpha_i) \\ \sin(\alpha_i) \end{bmatrix} \end{bmatrix} \begin{bmatrix} a \end{bmatrix} = \begin{bmatrix} T \\ 0 \end{bmatrix} .$$

Applying the conjugate transpose of the matrix on the left yields the normal equations

$$\begin{bmatrix} 1 + \lambda \end{bmatrix} \begin{bmatrix} a \end{bmatrix} = \begin{bmatrix} n(\boldsymbol{\alpha}) \end{bmatrix} .$$

The coefficient can be determined by inverting the matrix to obtain

$$\begin{bmatrix} a \end{bmatrix} = \begin{bmatrix} 1 + \lambda \end{bmatrix}^{-1} \begin{bmatrix} n(\boldsymbol{\alpha}) \end{bmatrix} = \frac{\begin{bmatrix} n(\boldsymbol{\alpha}) \end{bmatrix}}{(1 + \lambda)} . \tag{25}$$

Remark 1. The ALS algorithm solves a least-squares problem at each step, and so always maintains a at its optimal value. Line-search algorithms do not maintain optimal a, but could be modified to optimize a before evaluating the error function; it appears that the additional computational cost to do so would be negligible.

3.3. Error, gradient, and Hessian as a function of the angles. By making the scalar variable fast as in subsection 3.2 we eliminate a. Since the linear least-squares fitting to determine a is an orthogonal projection, we then know

$$E_\lambda(G_1) = \|T\|^2 - \left(\|G_1\|^2 + \lambda a^2\right) = 1 - \left(\|G_1\|^2 + \lambda a^2\right) , \tag{26}$$

recalling that $\|T\| = \|(3)\| = 1$. Evaluating directly and then using (25), we have

$$\|G_1\|^2 + \lambda a^2 = a^2(1 + \lambda) = \frac{n^2(\boldsymbol{\alpha})}{1 + \lambda} . \tag{27}$$

Therefore, combining (26) and (27), we obtain that

$$E_\lambda(G_1) = 1 - \frac{n^2(\boldsymbol{\alpha})}{1 + \lambda} . \tag{28}$$

We can compute the gradient and Hessian of $E_\lambda(G_1)$ directly as

$$\frac{\partial}{\partial \alpha_j} E_\lambda(G_1) = \frac{\partial}{\partial \alpha_j}(28) = -\frac{2n(\boldsymbol{\alpha}) n_j(\boldsymbol{\alpha})}{1 + \lambda} \quad \text{and} \tag{29}$$

$$\frac{\partial^2}{\partial \alpha_j \partial \alpha_k} E_\lambda(G_1) = \frac{\partial}{\partial \alpha_k}(29) = \frac{-2\left(n_k(\boldsymbol{\alpha}) n_j(\boldsymbol{\alpha}) + n(\boldsymbol{\alpha}) n_{jk}(\boldsymbol{\alpha})\right)}{1 + \lambda} . \tag{30}$$

4. Analysis with symmetric T and G_1. In this section we assume $\boldsymbol{\phi} = \phi\mathbf{1}$ and $\boldsymbol{\alpha} = \alpha\mathbf{1}$, which make T and G_1 symmetric under permutations of directions. To avoid T becoming rank one, we require $\phi \neq 0$ and so $\phi \in (0, \pi/2]$.

If $\phi_j = \phi_k$ and $\alpha_j = \alpha_k$, then $\frac{\partial}{\partial\alpha_j} E_\lambda(G_1) = \frac{\partial}{\partial\alpha_k} E_\lambda(G_1)$, so the equalities of the variables are preserved under the gradient flow. Thus the states with such symmetries are invariant sets. In subsection 4.2 we apply the method from subsection 2.2 to analyze the stability with respect to gradient flow of the invariant symmetric set with $\boldsymbol{\alpha} = \alpha\mathbf{1}$. In subsection 4.3 we further assume $z = \pm 1$ and analyze the stability of the symmetric angle $\alpha = \phi/2$ within the set with $\boldsymbol{\alpha} = \alpha\mathbf{1}$, as well as its properties as a stationary point in the full parameter space. In subsection 4.4 we provide visualizations of the approximation properties and interpret them for their implications for approximation algorithms. In subsection 4.5 we show by numerical experiments that the candidate swamps we identified do indeed result in swamps.

4.1. Preliminaries. The auxiliary functions in subsection 3.1 become

$$n(\boldsymbol{\alpha}) = \frac{\cos^d(\alpha) + z\cos^d(\alpha - \phi)}{(1 + 2z\cos^d(\phi) + z^2)^{1/2}}, \tag{31}$$

$$n_j(\boldsymbol{\alpha}) = \frac{-\sin(\alpha)\cos^{d-1}(\alpha) - z\sin(\alpha - \phi)\cos^{d-1}(\alpha - \phi)}{(1 + 2z\cos^d(\phi) + z^2)^{1/2}}, \tag{32}$$

$$\text{for } j \neq k \quad n_{jk}(\boldsymbol{\alpha}) = \frac{\sin^2(\alpha)\cos^{d-2}(\alpha) + z\sin^2(\alpha - \phi)\cos^{d-2}(\alpha - \phi)}{(1 + 2z\cos^d(\phi) + z^2)^{1/2}}, \quad \text{and} \tag{33}$$

$$n_{jj}(\boldsymbol{\alpha}) = -(31). \tag{34}$$

The first and second derivatives of E_λ can be computed as

$$\frac{\mathrm{d}}{\mathrm{d}\alpha} E_\lambda(G_1) = \mathbf{1}_d^* \nabla E_\lambda(G_1) = \sum_{j=1}^{d} (29) = d\frac{-2n(\boldsymbol{\alpha})n_j(\boldsymbol{\alpha})}{1 + \lambda} \quad \text{and} \tag{35}$$

$$\frac{\mathrm{d}^2}{\mathrm{d}\alpha^2} E_\lambda(G_1) = \mathbf{1}_d^* H \mathbf{1}_d = \sum_{j=1}^{d}\sum_{k=1}^{d} (30) = \frac{-2\left(d^2 n_j^2(\boldsymbol{\alpha}) + d(d-1)n(\boldsymbol{\alpha})n_{jk}(\boldsymbol{\alpha}) - dn^2(\boldsymbol{\alpha})\right)}{1 + \lambda}. \tag{36}$$

In a few cases we can explicitly locate maxima and minima.

Lemma 2. *For $\boldsymbol{\phi} = \phi\mathbf{1}$, if $z < 0$ or d is odd (or both), then a global maximum of $E_\lambda(G_1)$ occurs when $\boldsymbol{\alpha} = \alpha\mathbf{1}$ with*

$$\alpha = \arctan\left(\frac{(-z)^{-1/d} - \cos(\phi)}{\sin(\phi)}\right). \tag{37}$$

Proof. If $n(\boldsymbol{\alpha}) = 0$ then $E_\lambda(G_1) = 1 - n^2(\boldsymbol{\alpha})/(1 + \lambda) = 1$, which is a global maximum. Setting $n(\boldsymbol{\alpha})$ in (31) to 0 and manipulating yields

$$(-z)^{-1/d} = \frac{\cos(\alpha - \phi)}{\cos(\alpha)} = \cos(\phi) + \sin(\phi)\tan(\alpha),$$

which requires that $z < 0$ or d is odd (or both) so that we can compute $(-z)^{-1/d}$. Continuing to solve for α yields (37). ∎

If $z > 0$ and d is even, then $n(\boldsymbol{\alpha}) > 0$ for all α so we cannot locate the global maximum this way. To find other maxima or minima within the set $\boldsymbol{\alpha} = \alpha\mathbf{1}$, which may turn out to be saddles without this constraint, we set $\frac{\mathrm{d}}{\mathrm{d}\alpha}E_\lambda(G_1) = 0$ by setting $n_j(\boldsymbol{\alpha}) = 0$. If $\phi = \pi/2$, then $\alpha = 0$ and $\alpha = \pi/2$ solve $n_j(\boldsymbol{\alpha}) = 0$; they are minima since (36) yields $\frac{\mathrm{d}^2}{\mathrm{d}\alpha^2}E_\lambda(G_1) = \frac{2dn^2(\boldsymbol{\alpha})}{1+\lambda} > 0$. Otherwise, we have to solve

$$0 = (1 + \tan(\phi)\tan(\alpha))^{d-1}(\tan(\alpha) - \tan(\phi)) + z^{-1}\cos^{-d}(\phi)\tan(\alpha)\,,$$

which appears to be intractable.

4.2. Stability transverse to the flow on the symmetric set. As mentioned above, the gradient flow preserves the symmetric state. Given a small perturbation from a symmetric state, the flow may converge back toward the symmetric state or diverge away from it. To determine which is the case, we use the procedure in subsection 2.2.

Lemma 3. *For $\boldsymbol{\alpha} = \alpha\mathbf{1}$, the transverse Hessian $H^\perp(\boldsymbol{\alpha})$ has the single eigenvalue*

$$\frac{2(\cos^d(\alpha) + z\cos^d(\alpha-\phi))\left(\cos^{d-2}(\alpha) + z\cos^{d-2}(\alpha-\phi)\right)}{(1+\lambda)(1+2z\cos^d(\phi)+z^2)} \tag{38}$$

with multiplicity $d-1$.

Proof. Due to the symmetry, all the diagonal entries of the Hessian are the same and all the off-diagonal entries are the same. Using H_{11} to denote the diagonal value and H_{12} for the off-diagonal value, we can write

$$H(\boldsymbol{\alpha}) = (H_{11} - H_{12})I_d + H_{12}\mathbf{1}\,\mathbf{1}^*\,.$$

Conjugating with the Householder reflector U_{d1} from (5) yields

$$U_{d1}^* H(\boldsymbol{\alpha}) U_{d1} = (H_{11} - H_{12})I_d + dH_{12}\mathbf{e}_1\,\mathbf{e}_1^* = \begin{bmatrix} H_{11} + (d-1)H_{12} & 0 \\ 0 & (H_{11} - H_{12})I_{d-1} \end{bmatrix}. \tag{39}$$

Deleting the first row and column leaves $H^\perp(\boldsymbol{\alpha}) = (H_{11} - H_{12})I_{d-1}$, which has a single eigenvalue $(H_{11} - H_{12})$ with multiplicity $d-1$. To compute this eigenvalue, we start with (30) and insert (31), (32), (33), and (34), to obtain

$$\frac{-2\left((32)^2 - (31)^2\right)}{1+\lambda} - \frac{-2\left((32)^2 + (31)(33)\right)}{1+\lambda} = \frac{2(31)\left((31)+(33)\right)}{1+\lambda} = (38)\,. \qquad ∎$$

Theorem 4. *The gradient flow is transversely stable on the symmetric set except in the following cases:*

$$\begin{aligned}
&d \text{ is odd}, && z < 0\,, && \text{and} && \alpha \in [\alpha_+(d), \alpha_+(d-2)]\,; && (40)\\
&d \text{ is odd}, && 0 < z\,, && \text{and} && \alpha \in [\alpha_+(d-2), \alpha_+(d)]\,; \text{ and} && (41)\\
&d \text{ is even}, && z < 0\,, && \text{and} && \alpha \in [\alpha_-(d-2), \alpha_-(d)] \cup [\alpha_+(d), \alpha_+(d-2)]\,, &&
\end{aligned}$$

where

$$\alpha_\pm(k) = \arctan\left(\frac{\pm(-z)^{-1/k} - \cos(\phi)}{\sin(\phi)}\right).$$

On the given intervals in α the flow is transversely unstable except at the endpoints, where it is linearly neutral.

Proof. If d is even and $0 < z$, then (38) is always positive and so the flow is stable. Otherwise (38) changes sign when either of its factors do. When d is odd, we can set each factor of (38) to zero and solve the resulting equations to obtain sign changes at $\alpha_+(d)$ and $\alpha_+(d-2)$ with $\alpha_+(d) < \alpha_+(d-2)$ for $z < 0$ and $\alpha_+(d) > \alpha_+(d-2)$ for $z > 0$. Since (38) is positive at $\alpha = \pi/2$, its signs on the intervals are determined, and we obtain (40) and (41). When d is even and $z < 0$, we have the values from (40) and a second interval from the sign ambiguity. ∎

Note that the first factor in (38) is $n(\boldsymbol{\alpha})$, so when it is zero $E_\lambda(G_1) = 1$, which is the maximum error. The meaning of the second factor is not known.

4.3. Analysis with symmetric T and G_1 and with $z = \pm 1$. If $z = \pm 1$, then the target T is also symmetric with respect to a reflection about $\phi/2$ in each direction and multiplication by z. At the symmetry point $\phi/2$, we have

$$(31) \mapsto \qquad n(\boldsymbol{\phi}/2) = \frac{(1+z)\cos^d(\phi/2)}{(2+2z\cos^d(\phi))^{1/2}}, \tag{42}$$

$$(32) \mapsto \qquad n_j(\boldsymbol{\phi}/2) = \frac{-(1-z)\sin(\phi/2)\cos^{d-1}(\phi/2)}{(2+2z\cos^d(\phi))^{1/2}}, \quad \text{and} \tag{43}$$

$$(33) \mapsto \qquad n_{jk}(\boldsymbol{\phi}/2) = \frac{(1+z)\sin^2(\phi/2)\cos^{d-2}(\phi/2)}{(2+2z\cos^d(\phi))^{1/2}}, \tag{44}$$

so $n_j(\boldsymbol{\phi}/2) = 0$ when $z = 1$ and $n(\boldsymbol{\phi}/2) = n_{jk}(\boldsymbol{\phi}/2) = 0$ when $z = -1$.

Theorem 5. *If $z = -1$, then the symmetry point $\phi/2$ is a local maximum of $E_\lambda(G_1)$. If $z = 1$, then there is a pitchfork bifurcation at*

$$\phi_0 = 2\arcsin\left(d^{-1/2}\right) = 2\arctan\left((d-1)^{-1/2}\right), \tag{45}$$

with a local minimum at $\phi/2$ when $0 < \phi < \phi_0$ and a local maximum when $\phi_0 < \phi \le \pi/2$.

Proof. Inserting (42) and (43) in (35) yields $\frac{\mathrm{d}}{\mathrm{d}\alpha}E_\lambda(G_1)\big|_{\alpha=\phi/2} = 0$. Inserting (42)–(44) into (36) yields

$$\begin{aligned}\frac{\mathrm{d}^2}{\mathrm{d}\alpha^2}E_\lambda(G_1)\Big|_{\alpha=\phi/2} &= (1+z)^2\frac{-d\cos^{d-2}(\phi/2)\left(d\sin^2(\phi/2)-1\right)}{(1+\lambda)\left(1+z\cos^d(\phi)\right)} + (1-z)^2\frac{-d^2\sin^2(\phi/2)\cos^{2d-2}(\phi/2)}{(1+\lambda)\left(1+z\cos^d(\phi)\right)}.\end{aligned} \tag{46}$$

For $z = -1$, the first term in (46) drops out, and the result is strictly negative for $0 < \phi \le \pi/2$. Thus $E_\lambda(G_1)$ is always a local maximum and $\phi/2$ is a repelling/unstable fixed point.

For $z = 1$, the second term in (46) drops out, and the result is zero at (45). For $0 < \phi < \phi_0$, (46) is positive so $E_\lambda(G_1)$ is a local minimum and $\phi/2$ is an attracting/stable fixed point. Similarly, for $\phi_0 < \phi < \pi$, $E_\lambda(G_1)$ is a local maximum and $\phi/2$ is a repelling/unstable fixed point. At $\phi = \phi_0$, $\phi/2$ is a linearly neutral fixed point. ∎

For $z = -1$, the eigenvalue of the transverse Hessian given in (38) is zero at $\alpha = \phi/2$. Thus this maximum in the symmetric set is nonhyperbolic in the full space.

For $z = 1$, the eigenvalue of the transverse Hessian at $\alpha = \phi/2$ equals

$$\frac{4\cos^{2d-2}(\phi/2)}{(1+\lambda)(1+\cos^d(\phi))}. \tag{47}$$

Since this is positive, the maximum when $\phi_0 < \phi$ is a saddle in the full space and the minimum when $0 < \phi < \phi_0$ is a minimum in the full space. The eigenvalue of the Hessian within the symmetry set, which appears as $H_{11}+(d-1)H_{12}$ in (39), is also (46)/d. Taking the ratio yields

$$\frac{(46)/d}{(47)} = \frac{1 - d\sin^2(\phi/2)}{\cos^d(\phi/2)}. \tag{48}$$

For ϕ much larger than ϕ_0, we will have (48) < -1 and the analysis in subsection 2.3.1 shows an algorithm will escape from the saddle in one or two steps. For $\phi \to 0^+$, we have (48) $\to 1^-$ and the analysis in subsection 2.3.1 shows an algorithm will converge rapidly to the minimum. Since (48) $= 0$ at ϕ_0, there is a region of $\phi \approx \phi_0$ where escape from the saddle or convergence to the minimum will be very slow.

4.4. Visualization and interpretation. In this section we provide visualizations of the approximation. These allow us to explore more parameter choices than we did analytically. We then use the visualizations in combination with things we have proven in special cases to interpret phenomena that may cause approximation algorithms to converge slowly. Necessarily, these interpretations are not rigorous and so may later be overturned.

The first quantity we consider is the error itself, using the formulas from subsection 3.3. The second quantity is the estimated flow time, as determined in subsection 2.4. For visualization purposes we will map $s(\cdot) \in [0,\infty]$ from (16) to $\tilde{s}(G) \in [0,1]$ via

$$\tilde{s}(G) = \frac{s(G)}{1+s(G)} = \frac{E_\lambda(G)}{\|\nabla E_\lambda(G)\|_2^2 + E_\lambda(G)}. \tag{49}$$

The third quantity is the estimated algorithm time, as determined in subsection 2.4. Note that by Lemma 3 the transverse Hessian has a single eigenvalue μ, which will also play the role of η in (17). For visualization purposes we will map $v(\cdot) \in [0,\infty]$ from (17) to $\tilde{v}(G) \in [0,1]$ via

$$\tilde{v}(G) = \frac{v(G)}{1+v(G)} = \frac{E_\lambda(G)\mu(G)}{2\|\nabla E_\lambda(G)\|_2^2 + E_\lambda(G)\mu(G)}. \tag{50}$$

If $\mu(G) < 0$, then $\tilde{v}(G)$ is undefined and if $\mu(G) = 2$, then $\tilde{v}(G) = \tilde{s}(G)$.

The fourth quantity is the stability of the transverse Hessian from subsection 2.2, using the eigenvalue from Lemma 3. For visualization purposes, we map $h(\cdot) \in [-\infty,\infty]$ from (7) to $[0,1]$ and use

$$\tilde{h}(G) = \frac{1}{2}\left(1 - \frac{h(G)}{\sqrt{1+(h(G))^2}}\right) = \frac{1}{2}\left(1 - \frac{\mu(G)}{\sqrt{\|\nabla E_\lambda(G)\|_2^2 + (\mu(G))^2}}\right) \in [0,1]. \tag{51}$$

Values $\tilde{h}(G) \in [0,1/2)$ indicate stability and values $\tilde{h}(G) \in (1/2,1]$ indicate instability. When

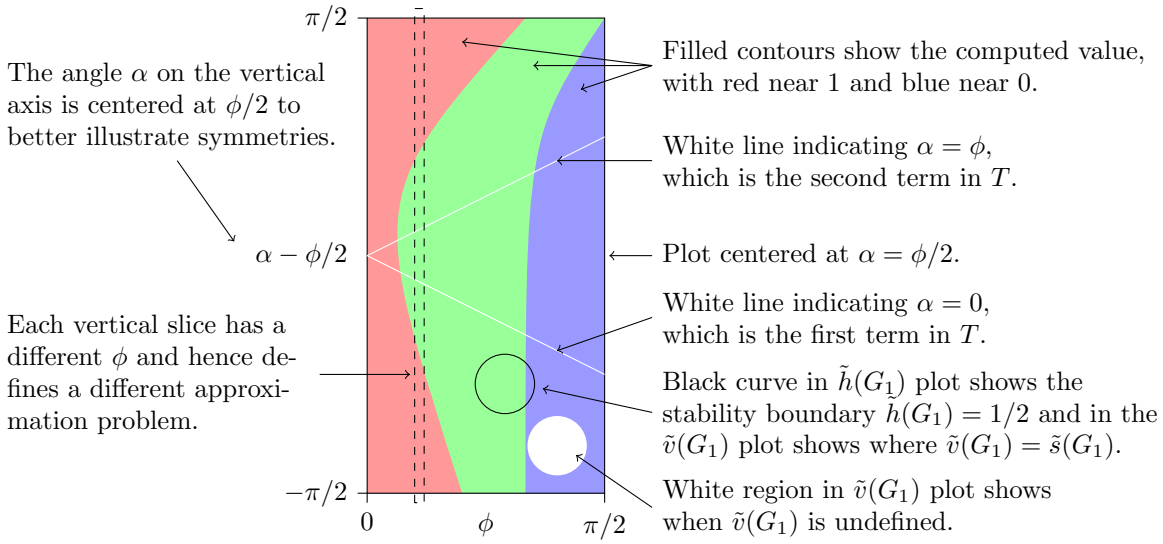

Figure 7. *Plotting format for the error $E_\lambda(G_1)$, estimated flow time $\tilde{s}(G_1) =$ (49), estimated algorithm time $\tilde{v}(G_1) =$ (50), and stability $\tilde{h}(G_1) =$ (51) for a fixed value of d, z, and λ.*

$\|\nabla E_\lambda(G)\|_2 \gg |\mu|$, then $\tilde{h}(G) \approx 1/2$, indicating that the (in)stability would not have as much effect.

We will plot each of these quantities as a function of ϕ and α while fixing d, z, and λ, using the format in Figure 7. In Figures 8 to 10 we show examples with $d \in \{5, 6, 30\}$, $z \in \{-1, -1/2, 1/2, 1\}$, and $\lambda = 0$. From our earlier analysis and the plots, we observe the following:

- When ϕ exceeds $\pi/2$ the case (d, z, ϕ) maps to $(d, (-1)^d z, \pi - \phi)$. In the plots, for even d one takes the plot, rotates it by π, shifts it up by $\pi/2$, and appends it to itself on the right. For odd d one rotates the plot by π, shifts it up by $\pi/2$, and appends it to the $-z$ plot.
- The $d = 30$ case is qualitatively similar to the $d = 5$ and $d = 6$ cases, but the interesting features are compressed and a larger proportion of the parameter space has large error.
- Local minima are more prominent at larger ϕ.
- When $z = 1$, the gradient is small in a large neighborhood of the minima when $\phi \approx \phi_0$ from (45). This region corresponds to values of the ratio (48) near 0, which subsection 2.3.1 shows indicate minima to which algorithms will converge very slowly or saddles by which algorithms will pass very slowly. For $0 < z < 1$ the bifurcation value for ϕ increases and the small gradient region is transitory, rather than around a minimum. Depending on the starting point, a gradient-based algorithm may have slow convergence while it passes through this region and then better convergence near the minima. These are swamp features.
- The symmetric state $\boldsymbol{\alpha} = \alpha\mathbf{1}$ is unstable only when the error is large. For algorithms, the implication is that symmetry is naturally lost only in the beginning when the approximation is very poor. Otherwise the symmetric state is locally attracting.

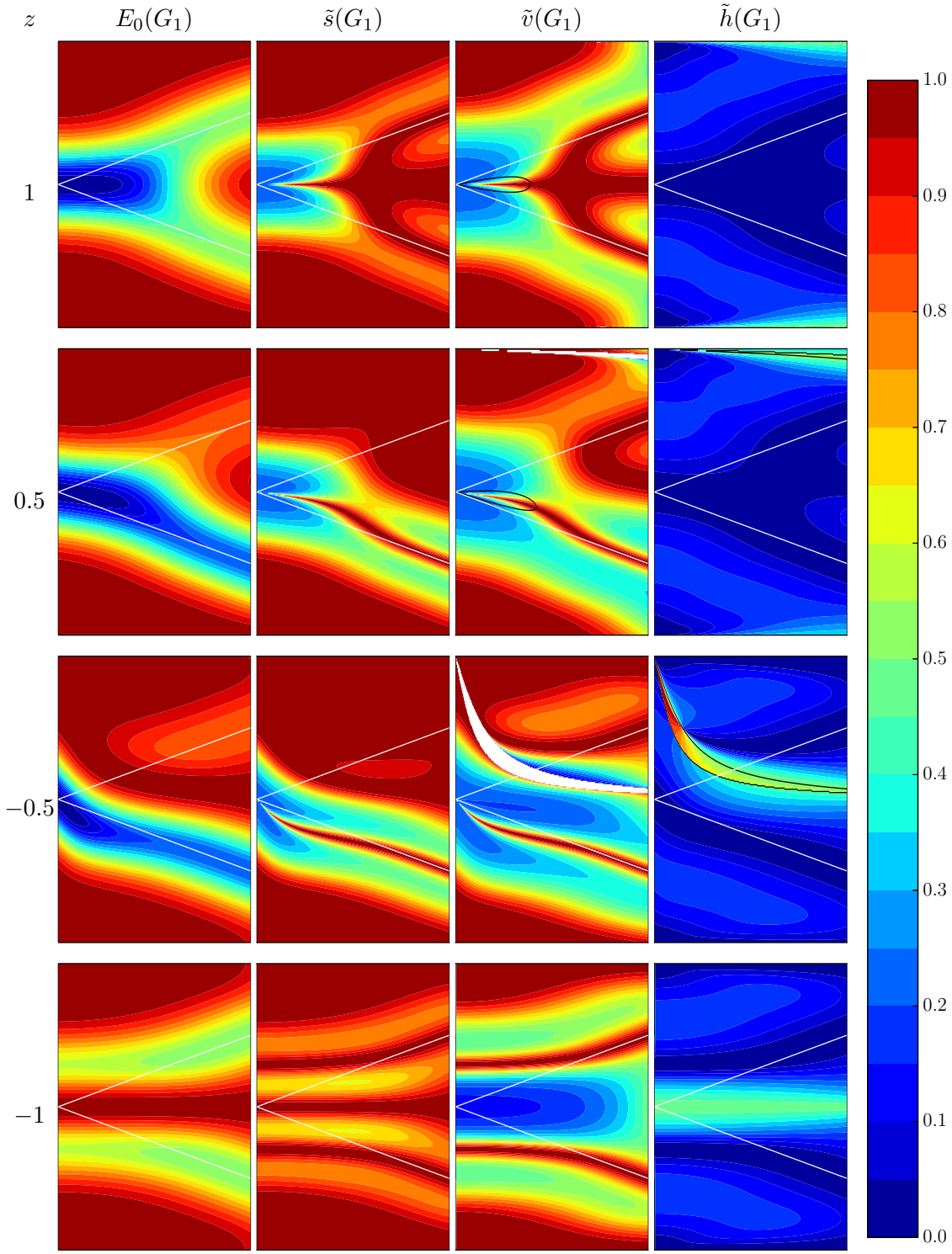

Figure 8. *Error $E_\lambda(G_1)$, estimated flow time $\tilde{s}(G_1)$, estimated algorithm time $\tilde{v}(G_1)$, and stability $\tilde{h}(G_1)$ landscapes for $d = 5$ and $\lambda = 0$ with z varying by row. Each subplot is formatted as in Figure 7.*

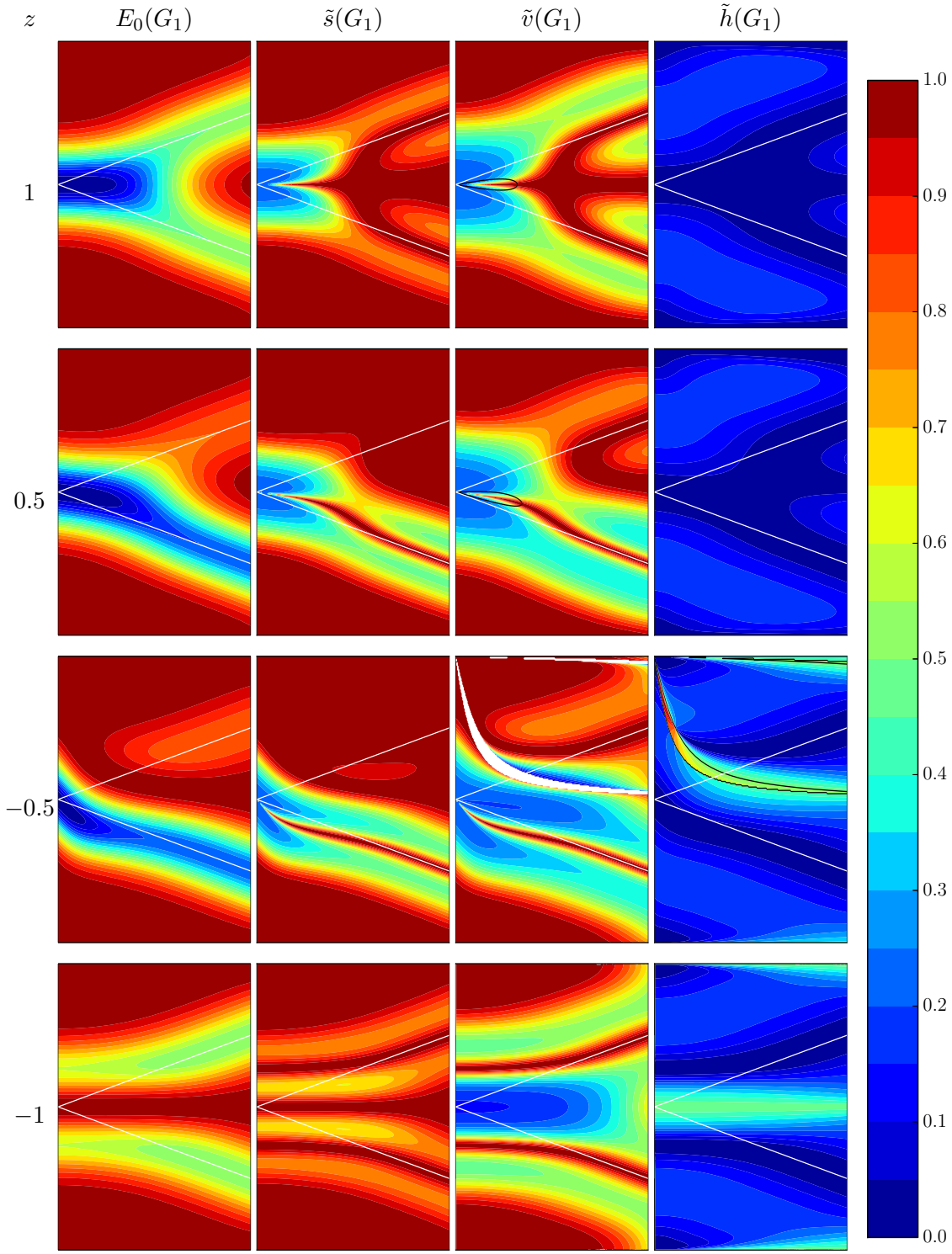

Figure 9. *Landscapes for $d = 6$ and $\lambda = 0$ in the same format as Figure 8.*

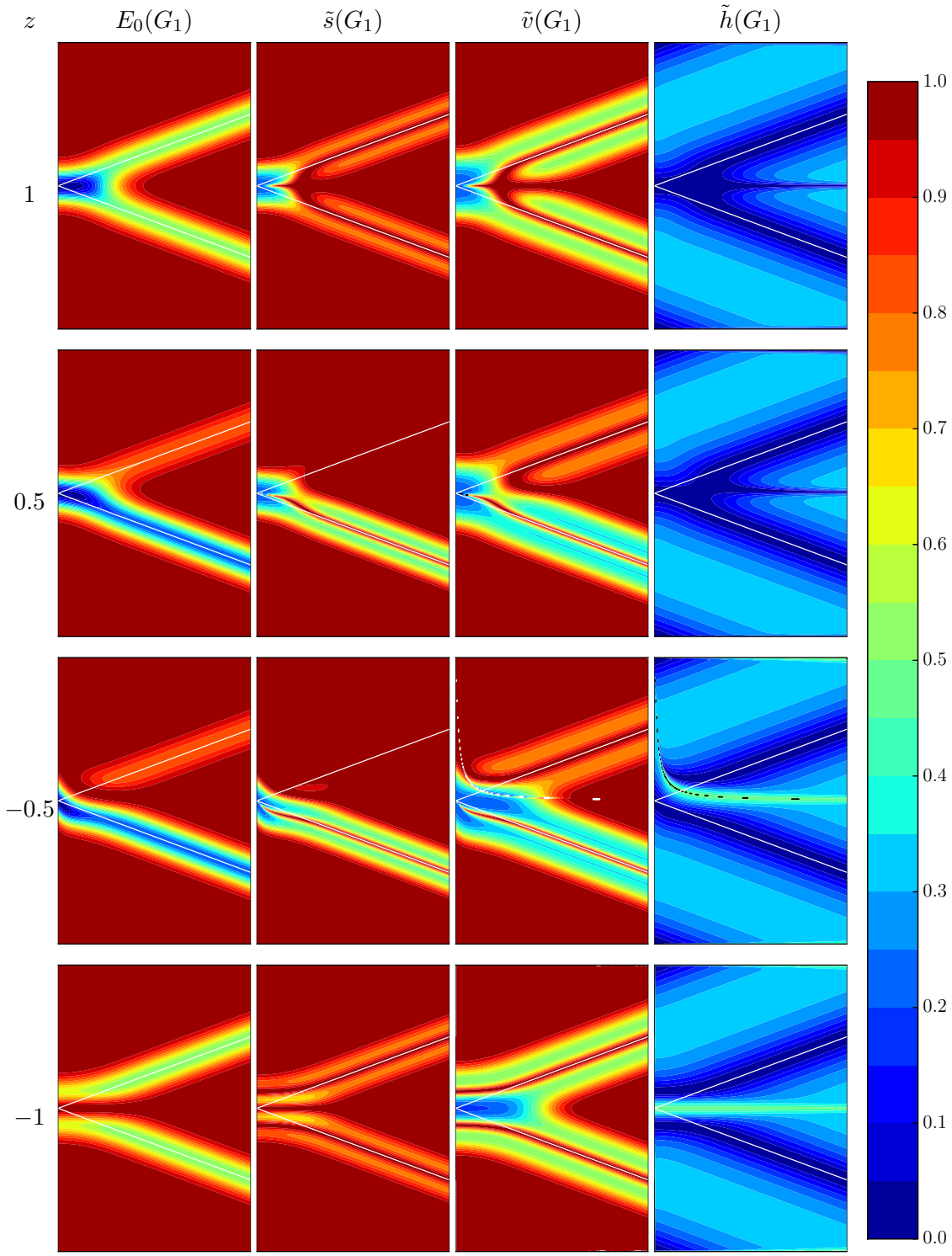

Figure 10. *Landscapes for $d = 30$ and $\lambda = 0$ in the same format as Figure 8. For $z = -0.5$, the unstable region is poorly resolved by the plotting program.*

- As suggested by the special case (47), the eigenvalue μ of the transverse Hessian is rarely more than 2, and thus the estimated algorithm time is usually smaller than the estimated flow time. The exception is when $z > 0$, ϕ is small, and α is near the global minimum. In some situations, such as the local maximum when $z = -1$ and $\alpha = \phi/2$, μ is small (in this case 0), which improves the situation.
- For $-1 < z < 0$, when ϕ is large there is a local minimum at $\alpha \approx \phi$; the symmetric state is stable there so this is also a local minimum in the full parameter space. As ϕ decreases there is a bifurcation point after which the symmetric state becomes unstable and so the minimum (in the symmetric set) is a saddle in the full space. This bifurcation point will appear again when we study asymmetric saddles in subsection 5.4 and we will locate it in Lemma 17.

Remark 6. We also produced plots for $\lambda = 1/2$, but they did not illustrate any new features, so we omit them.

4.5. Numerical validation of swamp candidates. The analysis in subsection 4.3 and the interpretation in subsection 4.4 indicate that setting $z = 1$ and $\phi = \phi_0 =$ (45) should result in a terminal swamp. Setting $(d, \lambda) = (6, 0)$ and starting point $\boldsymbol{\alpha} = \phi\mathbf{1}$, it takes 524 iterations of ALS before the change in error hits 10^{-9}, 1129 iterations for 10^{-10}, 2433 for 10^{-11}, etc. The error and change in error were plotted in Figure 1. Thus there indeed is sublinear convergence and a terminal swamp. Plots of $\boldsymbol{\alpha}$ show that $\boldsymbol{\alpha} \approx \alpha\mathbf{1}$ throughout the iterations.

The interpretation in subsection 4.4 also indicates that setting $0 < z < 1$ and ϕ slightly smaller than the transcritical bifurcation should result in a transient swamp. Fixing $(d, z, \lambda) = (6, 1/2, 0)$ and using starting point $\boldsymbol{\alpha} = \phi\mathbf{1}$, setting $\phi = 1.05503$ gives a transient swamp of length 760, where the change in error stays below 10^{-10} for many iterations. The error and change in error were plotted in Figure 2. Plots of $\boldsymbol{\alpha}$ show that $\boldsymbol{\alpha} \approx \alpha\mathbf{1}$ throughout the iterations.

5. Analysis with symmetric T and asymmetric G. In this section we assume $\boldsymbol{\phi} = \phi\mathbf{1}$ but that $\alpha_j \neq \alpha_i$ for some j, i, so T is symmetric but G is not. We study the stationary points of the gradient flow in (29) outside of the symmetric set. This section is based in part on the dissertation [45, Chapter 4]. In subsection 5.1 we consider the global maxima of the error. In subsection 5.2 we study the existence of stationary points and show that stationary points that are not global maxima must be of the form $\alpha_1 = \cdots = \alpha_m \neq \alpha_{m+1} = \cdots = \alpha_d$ and satisfy certain other conditions. In subsection 5.3 we consider the Hessian and show that these conditions imply that the stationary points are saddles (not minima or nonglobal maxima). In subsection 5.4 we analyze the saddle points within the framework of subsection 2.3.1 to determine when they would cause an algorithm to progress slowly; for $z < 0$ we find a non-hyberbolic stationary point at a nonzero angle and many fairly bad saddles at smaller angles.

Remark 7. It was already known that the global minimum of the error of approximation of a symmetric tensor of any rank by a rank-1 tensor is achieved by a symmetric rank-1 tensor; see [43], [14, Chapter 7], and [22, Corollary 4.2]. Our results in subsection 5.3 are slightly stronger in that they exclude local minima and local (nonglobal) maxima, but they apply only to symmetric rank-2 tensors.

5.1. Maximum points of the error. Global maxima of $E_\lambda(G_1)$ are plentiful and easy to describe. Compare with the case when $\boldsymbol{\alpha} = \alpha\mathbf{1}$ in Lemma 2.

Lemma 8. *When $\boldsymbol{\phi} = \phi\mathbf{1}$, the global maxima of $E_\lambda(G_1)$ occur when either*

$$\alpha_j = \pi/2 = \phi - \alpha_k \quad \text{for some } j \neq k \quad \text{or} \tag{52}$$

$$\alpha_d = \arctan\left(-\cot(\phi) - \frac{1}{z\sin(\phi)} \prod_{i=1}^{d-1} \frac{\cos(\alpha_i)}{\cos(\alpha_i - \phi)} \right). \tag{53}$$

Proof. If $n(\boldsymbol{\alpha}) = 0$, then $E_\lambda(G_1) = 1 - n^2(\boldsymbol{\alpha})/(1+\lambda) = 1$, which is a global maximum. Since $n(\boldsymbol{\alpha})$ in (21) is a sum of two products, there are two types of solutions to $n(\boldsymbol{\alpha}) = 0$.

If one product is zero, then both must be zero. Therefore a factor in each must be zero, which yields (52). The values for α_i for $k \neq i \neq j$ are unconstrained.

If neither product is zero, then we can manipulate $n(\boldsymbol{\alpha}) = 0$ to isolate α_d as

$$-\frac{1}{z}\prod_{i=1}^{d-1}\frac{\cos(\alpha_i)}{\cos(\alpha_i-\phi)} = \frac{\cos(\alpha_d-\phi)}{\cos(\alpha_d)} = \cos(\phi) + \tan(\alpha_d)\sin(\phi)\,.$$

Solving for α_d yields (53). ∎

5.2. Existence of nonsymmetric stationary points. To obtain the stationary points of the gradient flow, we set

$$(29) = \frac{\partial}{\partial\alpha_j} E_\lambda(G_1) = -\frac{2n(\boldsymbol{\alpha})n_j(\boldsymbol{\alpha})}{1+\lambda} = 0\,. \tag{54}$$

If $n(\boldsymbol{\alpha}) = 0$, then the point is a global maximum, as discussed in subsection 5.1; it may be symmetric or nonsymmetric. We therefore assume $n(\boldsymbol{\alpha}) \neq 0$ and study when $n_j(\boldsymbol{\alpha}) = 0$ for all j. First, in Theorem 9, we show that nonsymmetric stationary points must have a certain structure. Second, in Lemma 10, we derive a necessary and sufficient condition for points with this structure to actually be stationary points. Third, in Theorem 11, we determine when solutions to this condition, and thus stationary points, exist.

Theorem 9. *If $z \neq 0$, $\phi \in (0, \pi/2]$, $\alpha_i \in (-\pi/2, \pi/2]$ for all i, $n(\boldsymbol{\alpha}) \neq 0$, $n_i(\boldsymbol{\alpha}) = 0$ for all i, and $\alpha_1 \neq \alpha_k$ for some k, then, up to permutation of the directions,*

$$\alpha_1 = \cdots = \alpha_m \neq \alpha_{m+1} = \alpha_{m+2} = \cdots = \alpha_d \quad \text{for some } 1 \leq m \leq d/2 \text{ and} \tag{55}$$

$$\alpha_d = \frac{\pi}{2} + \phi - \alpha_1 + n\pi \quad \text{for some integer } n \text{ such that } \alpha_d \in (-\pi/2, \pi/2]. \tag{56}$$

If $\pi/2 \in \{\alpha_1, \alpha_d\}$, then $m = 1$, $\alpha_1 = \pi/2$, and $\alpha_d = \phi$.

Proof. By (22), we know that

$$0 = \|T\| n_j(\boldsymbol{\alpha}) = -\sin(\alpha_j)\prod_{i\neq j}\cos(\alpha_i) - z\sin(\alpha_j - \phi)\prod_{i\neq j}\cos(\alpha_i - \phi)\,. \tag{57}$$

Select k such that $\alpha_1 \neq \alpha_k$, multiply the n_k version of (57) by $\sin(\alpha_1-\phi)\cos(\alpha_k-\phi)$, multiply the n_1 version of (57) by $\sin(\alpha_k-\phi)\cos(\alpha_1-\phi)$, and subtract to obtain

(58)
$$0 = \left(\prod_{i\neq 1,\neq k} \cos(\alpha_i)\right) \times(\sin(\alpha_1)\cos(\alpha_k)\sin(\alpha_k-\phi)\cos(\alpha_1-\phi) - \sin(\alpha_k)\cos(\alpha_1)\sin(\alpha_1-\phi)\cos(\alpha_k-\phi)).$$

Direct expansion using sum-of-angles formulas shows that (58) is the same as

$$0 = \sin(\phi)\sin(\alpha_k-\alpha_1)\cos(\alpha_k+\alpha_1-\phi)\prod_{i\neq 1,\neq k}\cos(\alpha_i). \tag{59}$$

By our assumption $\phi \in (0,\pi/2]$ we know $\sin(\phi)\neq 0$ and by our assumption $\alpha_i \in (-\pi/2,\pi/2]$ for all i and $\alpha_1 \neq \alpha_k$ we know $\sin(\alpha_k-\alpha_1)\neq 0$.

We now split into a couple of cases. First suppose $\alpha_i \neq \pi/2$ for all i. Then (59) requires $\cos(\alpha_k+\alpha_1-\phi)=0$, which implies

$$\alpha_k = \frac{\pi}{2}+\phi-\alpha_1+n\pi \quad \text{for some integer } n \text{ such that } \alpha_k\in(-\pi/2,\pi/2]. \tag{60}$$

If there exists $p\neq k$ such that $\alpha_p \neq \alpha_1$, then we can subtract the corresponding equations (60) to obtain $\alpha_k-\alpha_p = n\pi$ for some integer n. Since $\alpha_i\in(-\pi/2,\pi/2]$ for all i, we therefore have $\alpha_p=\alpha_k$. If there exists $p\neq 1$ such that $\alpha_p\neq\alpha_k$, then we can switch the roles of 1 and k and conclude $\alpha_p=\alpha_1$. Thus there can only be two different values among the angles $\alpha_1,\dots,\alpha_d$. For convenience, we permute them so that repetitions of the least-frequent angle are first and so we have

$$\alpha_1=\cdots=\alpha_m\neq\alpha_{m+1}=\alpha_{m+2}=\cdots=\alpha_d \quad \text{for some } 1\leq m\leq d/2. \tag{61}$$

Under the additional assumption $\alpha_i\neq\pi/2$ for all i, we have thus proven (55) and by applying (60) have proven (56).

Now suppose $\alpha_p=\pi/2$ for exactly one p; by permuting we may assume $p=1$. Then (59) requires $\cos(\alpha_k+\pi/2-\phi)=0$, which implies $\alpha_k=\phi$ for all $k\neq 1$. Thus (55) and (56) hold in this case as well.

Finally suppose $\alpha_p=\alpha_q=\pi/2$ for some $p\neq q$. Then (57) reduces to

$$0 = \|T\|n_j(\boldsymbol{\alpha}) = z\sin(\alpha_j-\phi)\prod_{i\neq j}\cos(\alpha_i-\phi) \tag{62}$$

for all j. In order for (62) to hold for all j, we must have $\alpha_k=\phi-\pi/2$ for at least two values of k. However, having $\alpha_p=\pi/2$ and $\alpha_k=\phi-\pi/2$ makes $n(\boldsymbol{\alpha})=0$, which contradicts our assumption. ∎

From this point on, most results depend on the value of m in (55), which is the number of times the least frequent of the two angles occur at the nonsymmetric stationary point, and thus satisfies $1\leq m\leq d/2$.

Table 2

The existence of nonsymmetric stationary points of the gradient flow under the conditions of Theorem 11. If d is even and $m = d/2$, then the two-solutions cases correspond to one distinct solution and then switching α_1 with α_d.

	m odd		m even	
	$-1 \le z < 0$	$0 < z \le 1$	$-1 \le z < 0$	$0 < z \le 1$
d odd	at least 1	at least 1	at least 1	at least 1
d even	at least 2	do not exist	sometimes exist	at least 2

Lemma 10. *If $z \neq 0$, $\phi \in (0, \pi/2]$, $\alpha_i \in (-\pi/2, \pi/2]$ for all i, $n(\boldsymbol{\alpha}) \neq 0$, (55), and (56), then $n_i(\boldsymbol{\alpha}) = 0$ for all i if and only if*

$$0 = \cos^{m-1}(\alpha_1)\sin^{d-m-1}(\alpha_1 - \phi) + z\cos^{m-1}(\alpha_1 - \phi)\sin^{d-m-1}(\alpha_1)\,. \tag{63}$$

Proof. Due to (55), we need only consider $n_1(\boldsymbol{\alpha})$ and $n_d(\boldsymbol{\alpha})$. Plugging (61) into (57) for $j = 1$ and $j = d$ yields

$$0 = -\sin(\alpha_1)\cos^{m-1}(\alpha_1)\cos^{d-m}(\alpha_d) - z\sin(\alpha_1 - \phi)\cos^{m-1}(\alpha_1 - \phi)\cos^{d-m}(\alpha_d - \phi) \tag{64}$$

and

$$0 = -\sin(\alpha_d)\cos^{m}(\alpha_1)\cos^{d-m-1}(\alpha_d) - z\sin(\alpha_d - \phi)\cos^{m}(\alpha_1 - \phi)\cos^{d-m-1}(\alpha_d - \phi)\,. \tag{65}$$

From (56) we have

$$\cos(\alpha_d - \phi) = -\sin(\alpha_1)\,, \qquad \cos(\alpha_d) = -\sin(\alpha_1 - \phi)\,, \tag{66}$$

$$\sin(\alpha_d - \phi) = -\cos(\alpha_1)\,, \quad \text{and} \qquad \sin(\alpha_d) = -\cos(\alpha_1 - \phi)\,, \tag{67}$$

so we can eliminate α_d in (64) and (65) and obtain

$$0 = \sin(\alpha_1)\sin(\alpha_1 - \phi) \times\Big(\cos^{m-1}(\alpha_1)\sin^{d-m-1}(\alpha_1 - \phi) + z\cos^{m-1}(\alpha_1 - \phi)\sin^{d-m-1}(\alpha_1)\Big) \quad \text{and} \tag{68}$$

$$0 = \cos(\alpha_1)\cos(\alpha_1 - \phi) \times\Big(\cos^{m-1}(\alpha_1)\sin^{d-m-1}(\alpha_1 - \phi) + z\cos^{m-1}(\alpha_1 - \phi)\sin^{d-m-1}(\alpha_1)\Big)\,. \tag{69}$$

The condition (63) implies both (68) and (69), so one direction of the lemma is proven. If $\phi \neq \pi/2$, then $0 = \sin(\alpha_1)\sin(\alpha_1 - \phi)$ and $0 = \cos(\alpha_1)\cos(\alpha_1 - \phi)$ cannot simultaneously hold, so at least one of (68) or (69) implies (63). If $\phi = \pi/2$, then $0 = \sin(\alpha_1)\sin(\alpha_1 - \phi)$ and $0 = \cos(\alpha_1)\cos(\alpha_1 - \phi)$ simultaneously hold only for $\alpha_1 \in \{0, \pi/2\}$, but (56) then implies $\alpha_1 = \alpha_d$, which violates (55). ∎

Theorem 11. *If $z \neq 0$, $\phi \in (0, \pi/2]$, $\alpha_i \in (-\pi/2, \pi/2]$ for all i, $n(\boldsymbol{\alpha}) \neq 0$, (55), and (56), then solutions to $n_i(\boldsymbol{\alpha}) = 0$ for all i (i.e., stationary points) exist according to the cases in Table 2. For the case of d even, m even, and $z < 0$, such stationary points exist for*

$$z \in \left[\max\left\{-1, -\max_{\phi < x < \pi/2} \frac{\cos^{m-1}(x)\sin^{d-m-1}(x - \phi)}{\cos^{m-1}(x - \phi)\sin^{d-m-1}(x)}\right\}, 0\right)\,, \tag{70}$$

which is nonempty when $\phi \in (0, \pi/2)$. For $m = 1$ there is never such a stationary point with $\alpha_1 \in \{0, \phi\}$ and for $m > 1$ there is never such a stationary point with $\alpha_1 \in \{\phi - \pi/2, 0, \phi, \pi/2\}$.

Proof. By Lemma 10 we need only determine whether or not (63) has solutions. Let $f(\alpha_1)$ denote the right-hand side of (63), so we need to determine if f has roots in $(-\pi/2, \pi/2]$. Since f is continuous, we can prove existence by showing it has different sign at two points and applying the intermediate value theorem. Plugging in, we have

$$f(-\pi/2) = z\sin^{m-1}(\phi)(-1)^{d-2}, \qquad f(0) = (-1)^{d-m-1}\sin^{d-m-1}(\phi),$$
$$f(\phi) = z\sin^{d-m-1}(\phi), \quad \text{and} \qquad f(\pi/2) = z\sin^{m-1}(\phi).$$

If d is odd, then $f(-\pi/2)f(\pi/2) = -z^2\sin^{2m-2}(\phi) < 0$ so a root exists. If d is even, m is odd, and $z < 0$, then $f(-\pi/2)f(0) = z\sin^{2d-m-3}(\phi) < 0$ and $f(0)f(\phi) = z\sin^{2d-2m-2}(\phi) < 0$ so at least 2 roots exist. If d is even, m is even, and $z > 0$, then $f(-\pi/2)f(0) = -z\sin^{2d-m-3}(\phi) < 0$ and $f(0)f(\phi) = -z\sin^{2d-2m-2}(\phi) < 0$ so at least 2 roots exist.

If d is even, m is odd, and $z > 0$, then each individual factor in f is nonnegative, so $f(\alpha_1) = 0$ if and only if $0 = \cos(\alpha_1)\sin(\alpha_1 - \phi)$ and $0 = \sin(\alpha_1)\cos(\alpha_1 - \phi)$ simultaneously hold. For one of them to be zero we need $\alpha_1 \in \{\phi - \pi/2, 0, \phi, \pi/2\}$ but then plugging in shows the other is not zero. Thus in this case no roots exist. This argument also proves the last part of the theorem, that for $m = 1$ there is never a solution with $\alpha_1 \in \{0, \phi\}$ and for $m > 1$ there is never a solution with $\alpha_1 \in \{\phi - \pi/2, 0, \phi, \pi/2\}$, since those make one term in f be zero while leaving the other nonzero.

If d is even, m is even, and $z < 0$, then $f(\alpha_1) < 0$ for $\alpha_1 \in \{-\pi/2, \phi - \pi/2, 0, \phi, \pi/2\}$ and we have to work harder to find a point with $f(\alpha_1) > 0$. We first consider the case $\phi = \pi/2$ for which

$$f(\alpha_1) = (-1)^{d-m-1}\cos^{d-2}(\alpha_1) + z\sin^{d-2}(\alpha_1).$$

Both terms are nonpositive and they cannot be simultaneously zero, so there are no roots. Now we suppose $\phi \neq \pi/2$. On the interval $\alpha_1 \in (-\pi/2, \phi - \pi/2)$ and $\alpha_1 \in (0, \phi)$, both terms in $f(\alpha_1)$ are strictly negative so $f(\alpha_1) < 0$ and so does not have a root. On the interval $\alpha_1 \in (\phi, \pi/2)$, the first term in $f(\alpha_1)$ is strictly positive and the second is strictly negative, so when $|z|$ is sufficiently small we have $f(\alpha_1) > 0$ and so a root exists. We can write the condition on z explicitly as (70); when z is in the interior of the interval there are at least 2 roots and when z equals the left boundary there may only be one. On the interval $\alpha_1 \in (\phi - \pi/2, 0)$, the first term in $f(\alpha_1)$ is negative and the second is positive, so when $|z|$ is sufficiently large we have $f(\alpha_1) > 0$ and so a root exists. The condition on z is

$$z < \max_{\phi - \pi/2 < x < 0} \frac{\cos^{m-1}(x)\sin^{d-m-1}(x - \phi)}{\cos^{m-1}(x - \phi)\sin^{d-m-1}(x)}. \tag{71}$$

However, when $x \in (\phi - \pi/2, 0)$ we have

$$1 < \frac{\cos(x)}{\cos(x - \phi)} \qquad \text{and} \qquad \frac{\sin(x - \phi)}{\sin(x)} < -1,$$

so (71) implies $z < -1$, which violates our assumption. ∎

5.3. Analysis of the Hessian matrix at the nonsymmetric stationary points. In this section we show that the nonsymmetric stationary points that are not global maxima must be saddles (and so not minima or nonglobal maxima). In Lemma 12 we use the results of the previous section to compute the entries of the Hessian and determine its structure. In Lemma 13 we use unitary transformations to make the eigenvalues of the Hessian transparent. In Theorem 14 we show that in all cases at least one eigenvalue is negative and thus there is a saddle.

Lemma 12. *If $z \neq 0$, $\phi \in (0, \pi/2]$, $\alpha_i \in (-\pi/2, \pi/2]$ for all i, not all α_i are the same, $n(\boldsymbol{\alpha}) \neq 0$, and $n_i(\boldsymbol{\alpha}) = 0$ for all i, then, up to permutation of the directions, the Hessian at $\boldsymbol{\alpha}$ can be written as*

$$H = \frac{2(n(\boldsymbol{\alpha}))^2}{1+\lambda} \begin{bmatrix} (1-\Theta^{-1})I_m + \Theta^{-1}\mathbf{1}_m\mathbf{1}_m^* & \mathbf{1}_m\mathbf{1}_{d-m}^* \\ \mathbf{1}_{d-m}\mathbf{1}_m^* & (1-\Theta)I_{d-m} + \Theta\mathbf{1}_{d-m}\mathbf{1}_{d-m}^* \end{bmatrix}, \tag{72}$$

$$\text{where} \quad \Theta = \cot(\alpha_1)\cot(\alpha_1 - \phi)\,. \tag{73}$$

If $m = 1$, then $1 \neq \Theta$ and Θ^{-1} cancels out. If $m > 1$, then $1 \neq \Theta \neq 0 \neq \Theta^{-1}$.

Proof. The Hessian depends on $n(\boldsymbol{\alpha})$ and $n_{jk}(\boldsymbol{\alpha})$, so we need to compute them. The procedure is to

1. start with the original formula (21) for $n(\boldsymbol{\alpha})$ and (23) or (24) for $n_{jk}(\boldsymbol{\alpha})$,
2. apply the structural constraint (55),
3. apply (56) in the form (66) and (67) to eliminate α_d,
4. apply (63) in the form

$$z = -\cos^{m-1}(\alpha_1)\sin^{d-m-1}(\alpha_1-\phi)\cos^{-m+1}(\alpha_1-\phi)\sin^{-d+m+1}(\alpha_1)$$

   to eliminate z, and
5. use trigonometric identities to simplify.

For $n(\boldsymbol{\alpha})$ we obtain

$$\begin{aligned} \|T\|n(\boldsymbol{\alpha}) &= \cos^m(\alpha_1)\cos^{d-m}(\alpha_d) + z\cos^m(\alpha_1-\phi)\cos^{d-m}(\alpha_d-\phi) \\ &= (-1)^{d-m}\cos^m(\alpha_1)\sin^{d-m}(\alpha_1-\phi) + (-1)^{d-m}z\cos^m(\alpha_1-\phi)\sin^{d-m}(\alpha_1) \\ &= (-1)^{d-m}\sin^{d-m-1}(\alpha_1-\phi)\cos^{m-1}(\alpha_1)\left(\cos(\alpha_1)\sin(\alpha_1-\phi) - \cos(\alpha_1-\phi)\sin(\alpha_1)\right) \\ &= (-1)^{d-m-1}\sin(\phi)\sin^{d-m-1}(\alpha_1-\phi)\cos^{m-1}(\alpha_1)\,. \end{aligned} \tag{74}$$

For $n_{jk}(\boldsymbol{\alpha})$ we have to split into cases based on the relationship of j, k, and m. When $j = k$, by (24) we have $\|T\|n_{jk}(\boldsymbol{\alpha}) = -\|T\|n(\boldsymbol{\alpha}) = -(74)$. For $j \leq m$, $k \leq m$, and $j \neq k$ we obtain

$$\|T\|n_{jk}(\boldsymbol{\alpha}) = (-1)^{d-m}\sin(\phi)\frac{\sin(\alpha_1)}{\cos(\alpha_1-\phi)}\sin^{d-m}(\alpha_1-\phi)\cos^{m-2}(\alpha_1)\,. \tag{75}$$

For $j > m$, $k > m$, and $j \neq k$ we obtain

$$\|T\|n_{jk}(\boldsymbol{\alpha}) = (-1)^{d-m}\sin(\phi)\frac{\cos(\alpha_1-\phi)}{\sin(\alpha_1)}\sin^{d-m-2}(\alpha_1-\phi)\cos^{m}(\alpha_1)\,. \tag{76}$$

For $j \le m$ and $k > m$, or $j > m$ and $k \le m$, we obtain

$$\|T\| n_{jk}(\boldsymbol{\alpha}) = (-1)^{d-m} \sin(\phi) \sin^{d-m-1}(\alpha_1 - \phi) \cos^{m-1}(\alpha_1) = -(74) = -\|T\| n(\boldsymbol{\alpha}) \,. \tag{77}$$

Starting from (30) and applying the assumption that $n_i(\boldsymbol{\alpha}) = 0$ for all i, the entries in the Hessian H are given by

$$H_{jk} = \frac{-2\left(n_k(\boldsymbol{\alpha}) n_j(\boldsymbol{\alpha}) + n(\boldsymbol{\alpha}) n_{jk}(\boldsymbol{\alpha})\right)}{1+\lambda} = \frac{-2n(\boldsymbol{\alpha}) n_{jk}(\boldsymbol{\alpha})}{1+\lambda} = \frac{2(n(\boldsymbol{\alpha}))^2}{1+\lambda} \frac{-n_{jk}(\boldsymbol{\alpha})}{n(\boldsymbol{\alpha})} \,. \tag{78}$$

To prove the lemma, it suffices to show that $-n_{jk}(\boldsymbol{\alpha})/n(\boldsymbol{\alpha})$ has the block structure described for (72). If $j = k$, by (24) we have $-n_{jk}(\boldsymbol{\alpha})/n(\boldsymbol{\alpha}) = 1$ as desired. For $j \le m$, $k \le m$, and $j \ne k$ we obtain

$$\frac{-n_{jk}(\boldsymbol{\alpha})}{n(\boldsymbol{\alpha})} = -\frac{(75)}{(74)} = \frac{\sin(\alpha_1)\sin(\alpha_1 - \phi)}{\cos(\alpha_1 - \phi)\cos(\alpha_1)} = \Theta^{-1} \,,$$

as desired. For $j > m$, $k > m$, and $j \ne k$ we obtain

$$\frac{-n_{jk}(\boldsymbol{\alpha})}{n(\boldsymbol{\alpha})} = -\frac{(76)}{(74)} = \frac{\cos(\alpha_1 - \phi)\cos(\alpha_1)}{\sin(\alpha_1)\sin(\alpha_1 - \phi)} = \Theta \,,$$

as desired. For $j \le m$ and $k > m$, or $j > m$ and $k \le m$, by (77) we have $-n_{jk}(\boldsymbol{\alpha})/n(\boldsymbol{\alpha}) = 1$ as desired.

By Theorem 11, if $m > 1$, then $\alpha_1 \in \{\phi - \pi/2, 0, \phi, \pi/2\}$ is never a solution, so $\Theta \ne 0$ and $\Theta^{-1} \ne 0$. If $\Theta = 1$, then $\Theta = (73) = 1$ yields $\cos(2\alpha_1 - \phi) = 0$, which implies $\alpha_1 = \phi/2 \pm \pi/4$. Then by (56) we have $\alpha_d = \phi/2 \pm \pi/4 = \alpha_1$. Since this contradicts (55), we know $\Theta \ne 1$. ∎

Lemma 13. *The Hessian (72) has eigenvalues* $2(n(\boldsymbol{\alpha}))^2(1+\lambda)^{-1}$ *times*

$$1 - \Theta^{-1} \quad \textit{with multiplicity } m-1, \tag{79}$$

$$1 - \Theta \quad \textit{with multiplicity } d-m-1, \textit{ and} \tag{80}$$

$$\textit{the eigenvalues of } \begin{bmatrix} 1 + (m-1)\Theta^{-1} & \sqrt{m(d-m)} \\ \sqrt{m(d-m)} & 1 + (d-m-1)\Theta \end{bmatrix} . \tag{81}$$

Proof. The Householder reflector U_{pq} defined in (5) is unitary, symmetric, and takes $\mathbf{1}_p$ to $\sqrt{p}\mathbf{e}_q$. We divide the leading factor out of H and conjugate with unitary matrices to obtain

$$\begin{aligned} &\begin{bmatrix} U_{mm} & 0 \\ 0 & I_{d-m} \end{bmatrix} \begin{bmatrix} I_m & 0 \\ 0 & U_{(d-m)1} \end{bmatrix} \frac{1+\lambda}{2(n(\boldsymbol{\alpha}))^2} H \begin{bmatrix} I_m & 0 \\ 0 & U_{(d-m)1} \end{bmatrix} \begin{bmatrix} U_{mm} & 0 \\ 0 & I_{d-m} \end{bmatrix} \\ &\quad = \begin{bmatrix} (1-\Theta^{-1})I_m + \Theta^{-1} m \mathbf{e}_m \mathbf{e}_m^* & \sqrt{m}\sqrt{d-m}\mathbf{e}_m \mathbf{e}_1^* \\ \sqrt{m}\sqrt{d-m}\mathbf{e}_1 \mathbf{e}_m^* & (1-\Theta)I_{d-m} + \Theta(d-m)\mathbf{e}_1 \mathbf{e}_1^* \end{bmatrix} \\ &\quad = \begin{bmatrix} (1-\Theta^{-1})I_{m-1} & 0 & 0 & 0 \\ 0 & 1 + \Theta^{-1}(m-1) & \sqrt{m}\sqrt{d-m} & 0 \\ 0 & \sqrt{m}\sqrt{d-m} & 1 + \Theta(d-m-1) & 0 \\ 0 & 0 & 0 & (1-\Theta)I_{d-m-1} \end{bmatrix} . \end{aligned} \tag{82}$$

Conjugation with unitary matrices preserves eigenvalues and by reading the eigenvalues off (82) the lemma is proven. ∎

Theorem 14. *If $z \neq 0$, $\phi \in (0, \pi/2]$, $\alpha_i \in (-\pi/2, \pi/2]$ for all i, $\alpha_1 \neq \alpha_k$ for some k, $n(\boldsymbol{\alpha}) \neq 0$, and $n_i(\boldsymbol{\alpha}) = 0$ for all i, then $\boldsymbol{\alpha}$ is a saddle point.*

Proof. Let $x_m \geq x_{m+1}$ denote the eigenvalues of (81). By computing the determinant and trace of (81), we have

$$x_m x_{m+1} = (2-d) + (d-m-1)\Theta + (m-1)\Theta^{-1} \quad \text{and} \tag{83}$$

$$x_m + x_{m+1} = 2 + (m-1)\Theta^{-1} + (d-m-1)\Theta\,. \tag{84}$$

If $\Theta < 0$, then each summand in (83) is negative, so $x_m > 0 > x_{m+1}$ and we have a saddle. If $\Theta > 0$, then one of $1 - \Theta^{-1}$ or $1 - \Theta$ is positive and the other is negative; if $m > 1$, then both $1 - \Theta^{-1}$ and $1 - \Theta$ are eigenvalues, so we have a saddle. If $0 \leq \Theta < 1$ and $m = 1$, then (83) becomes $x_1 x_2 = (2-d)(1-\Theta) < 0$ so $x_1 > 0 > x_2$ and we have a saddle. If $\Theta > 1$ and $m = 1$, then $1 - \Theta < 0$ so we still have a negative eigenvalue; by (83) and (84), we have $x_1 x_2 = (d-2)(\Theta - 1) > 0$ and $x_1 + x_2 = 2 + (d-2)\Theta > 0$, so $x_1 \geq x_2 > 0$ and we still have a saddle. By Lemma 12, $\Theta \neq 1$ and for $m > 1$ also $\Theta \neq 0$. ∎

5.4. Transit past saddles. In this section we analyze the saddles using the methods from subsection 2.3.1. In particular, we are interested in the ratio μ/η, where μ is the smallest (most negative) and η is the largest eigenvalue of the Hessian H. If $\mu/\eta \leq -1$, then an algorithm will escape in one or two steps; otherwise each iteration multiplies the distance by (at minimum) $(1 - \mu/\eta)/(1 + \mu/\eta)$ from (10).

In subsection 5.4.1 we consider the case $m = 1$. We can solve (63) for α_1 and from there determine complete information about the saddles. We show that for $z < 0$, saddles with $\mu/\eta \to 0^-$ exist as ϕ approaches a certain nonzero angle, whereas for $0 < z$ the ratio μ/η is bounded away from zero.

In subsection 5.4.2 we consider $m > 1$. We cannot solve (63) analytically, so we solve it numerically and then numerically determine μ/η. We observe that for $z < 0$ as ϕ decreases there are bifurcation points at which additional saddles are created. The worst case is still when $\mu/\eta \to 0^-$ near the same nonzero angle, but for smaller angles there can be a large number of bad saddles.

5.4.1. Saddles with $m = 1$. In Lemma 15 we determine the value of μ/η as a function of d and $\Theta = (73) = \cot(\alpha_1)\cot(\alpha_1 - \phi)$ and show that the worst case is when $\Theta \to 1$, which yields $\mu/\eta \to 0^-$. In Lemma 16 we determine the value(s) of α_1 that yield a saddle, as a function of d, ϕ, and z. Thus for any configuration (d, ϕ, z) we can determine μ/η. In Lemma 17 we show that $\Theta = 1$ corresponds to a symmetric, nonhyperbolic stationary point (saddle or possibly minimum) with $\mu/\eta = 0$ and show that for any d and $-1 < z < 0$ there exists $\phi \in (0, \pi/2)$ such that $\Theta = 1$. In Lemma 18 we show that $0 < z$ implies $\Theta \leq -1$ which by Lemma 15 implies $\mu/\eta < (2-d)/2 \leq -1/2$.

Lemma 15. *Under the conditions of Theorem 14, if $m = 1$, then on $-\infty < \Theta \leq -1$ the Hessian has $d - 1$ positive eigenvalues and 1 negative eigenvalue, and*

$$\frac{\mu}{\eta} = \frac{2(2-d)}{(2 + (d-2)\Theta) + \sqrt{(2 + (d-2)\Theta)^2 - 4(d-2)(\Theta - 1)}}, \tag{85}$$

Table 3
Analysis of cases for Lemma 15.

$\Theta \in$	$(-\infty, -1]$	$(-1, 1)$	$(1, \infty)$
$x_d = 1 - \Theta$ by (89)	$x_d > 0$ $x_1 x_2 < 0$ so $x_1 > 0 > x_2$		$x_d < 0$ $x_1 x_2 > 0$
by (90)	$(x_1 - x_d)(x_2 - x_d) \geq 0$ so $x_d \geq x_1 > x_2$	$(x_1 - x_d)(x_2 - x_d) < 0$ so $x_1 > x_d > x_2$	
by (91)			$x_1 + x_2 > 0$ so $x_1 \geq x_2 > 0$
Combined	$x_d \geq x_1 > 0 > x_2$	$x_1 > x_d > 0 > x_2$	$x_1 \geq x_2 > 0 > x_d$
μ/η	x_2/x_d	x_2/x_1	x_d/x_1

which is increasing and has value $(2-d)/2$ *at* $\Theta = -1$*; on* $-1 \leq \Theta < 1$ *the Hessian has* $d-1$ *positive eigenvalues and* 1 *negative eigenvalue, and*

$$\frac{\mu}{\eta} = \frac{(2 + (d-2)\Theta) - \sqrt{(2 + (d-2)\Theta)^2 - 4(d-2)(\Theta - 1)}}{(2 + (d-2)\Theta) + \sqrt{(2 + (d-2)\Theta)^2 - 4(d-2)(\Theta - 1)}}, \tag{86}$$

which is increasing and has value $(2-d)/2$ *at* $\Theta = -1$ *and limit* 0 *as* $\Theta \to 1^-$*; and on* $1 < \Theta < \infty$ *the Hessian has* 2 *positive eigenvalues and* $d-2$ *negative eigenvalues, and*

$$\frac{\mu}{\eta} = \frac{(2 + (d-2)\Theta) - \sqrt{(2 + (d-2)\Theta)^2 - 4(d-2)(\Theta - 1)}}{2(2-d)}, \tag{87}$$

which is decreasing and has limit 0 *as* $\Theta \to 1^+$.

Proof. When $m = 1$, we no longer have the eigenvalue $1 - \Theta^{-1}$ from (79) but still have the eigenvalue $x_d = 1 - \Theta$ from (80) with multiplicity $d-2$. By (81), x_1 and x_2 are eigenvalues of

$$\begin{bmatrix} 1 & \sqrt{d-1} \\ \sqrt{d-1} & 1 + (d-2)\Theta \end{bmatrix}, \tag{88}$$

from which we can compute

$$\det((88)) = x_1 x_2 = (d-2)(\Theta - 1), \tag{89}$$

$$\det((88) - (1-\Theta)I) = (x_1 - x_d)(x_2 - x_d) = (d-1)(\Theta^2 - 1), \quad \text{and} \tag{90}$$

$$\operatorname{trace}((88)) = x_1 + x_2 = 2 + (d-2)\Theta. \tag{91}$$

To determine μ/η, we need to determine the signs and order of the eigenvalues. We organize the cases in Table 3. We can directly compute the eigenvalues

$$x_1, x_2 = \frac{(2 + (d-2)\Theta) \pm \sqrt{(2 + (d-2)\Theta)^2 - 4(d-2)(\Theta - 1)}}{2} \tag{92}$$

and the required ratios x_2/x_d, x_2/x_1, and x_d/x_1. By (89) we have $x_1 x_2 = (2-d)x_d$, so we can simplify somewhat by writing $x_2/x_d = (2-d)/x_1$ and $x_d/x_1 = x_2/(2-d)$ to obtain (85)–(87).

The values of μ/η at $\Theta = -1$ are determined by plugging into (85) and (86). The limits of μ/η as $\Theta \to 1^{\pm}$ are obtained by plugging into (86) and (87). To determine the intervals of increase and decrease, we compute the derivative of μ/η. Since the computation is routine but long, we omit it. ∎

Lemma 16. *For $m = 1$, the saddles are located at*

$$\alpha_1 = \arctan\left(\frac{\sin(\phi)}{\cos(\phi) - S(-z)^{1/(d-2)}}\right), \tag{93}$$

where $S = 1$ if d is odd, $S = \pm 1$ both give saddles if d is even and $z < 0$, and no saddles exist (by Theorem 11) if d is even and $z > 0$.

Proof. For $m = 1$, (63) becomes

$$0 = \sin^{d-2}(\alpha_1 - \phi) + z\sin^{d-2}(\alpha_1),$$

which we can solve for z to obtain

$$z = -\left(\frac{\sin(\alpha_1 - \phi)}{\sin(\alpha_1)}\right)^{d-2} = -\left(\cos(\phi) - \sin(\phi)\cot(\alpha_1)\right)^{d-2}. \tag{94}$$

If $z > 0$ and d is even, then there is no solution to (94). Otherwise we note that $\cot(\alpha_1)$ is one-to-one and has range $(-\infty, \infty)$, so there is exactly one solution to (94) when d is odd and two solutions corresponding to signs ± 1 when d is even. To treat these cases together, let $S = 1$ for d odd and $S = \pm 1$ for d even. Solving (94) for α_1 yields (93). ∎

Lemma 17. *If $d > 2$ and $\phi \in (0, \pi/2)$, then*

$$z = -\left(\frac{\sin(\pi/4 - \phi/2)}{\sin(\pi/4 + \phi/2)}\right)^{d-2} \in (-1, 0) \tag{95}$$

and (d, z, ϕ) yields a nonhyperbolic, symmetric stationary point located at $\boldsymbol{\alpha} = (\phi/2 + \pi/4)\mathbf{1}$ with $d - 1$ zero eigenvalues and a single positive eigenvalue. Moreover, (95) maps $(0, \pi/2)$ onto $(-1, 0)$, so there is such a point for every $z \in (-1, 0)$.

Proof. In Lemma 12 we excluded $\Theta = 1$ because it made $\boldsymbol{\alpha} = \alpha\mathbf{1}$ with $\alpha = \phi/2 \pm \pi/4$. From Lemma 15 we now see that when $\Theta = 1$ there is a symmetric stationary point with $d-1$ zero eigenvalues (x_2 and x_d) and a single positive eigenvalue (x_1). Within the symmetric set this point is a local minimum. Inserting $\alpha = \phi/2 \pm \pi/4$ into (94) yields

$$z = -\left(\frac{\sin(\phi/2 \pm \pi/4 - \phi)}{\sin(\phi/2 \pm \pi/4)}\right)^{d-2} = -\left(\frac{\sin(\pi/4 \mp \phi/2)}{\sin(\pi/4 \pm \phi/2)}\right)^{d-2}.$$

By our assumption $|z| \le 1$ we must choose $\pm \mapsto +$ and by our assumption $\phi \in (0, \pi/2)$ we must have $-1 < z < 0$. Since (95) $\to -1^+$ as $\phi \to 0^+$ and (95) $\to 0^-$ as $\phi \to (\pi/2)^-$, we see (95) maps $(0, \pi/2)$ onto $(-1, 0)$. ∎

Lemma 18. *If $m = 1$, and $0 < z \leq 1$, then $\Theta \leq -1$, and thus by Lemma 15 $\mu/\eta \leq (2-d)/2$.*

Proof. By Lemma 16, we know d must be odd and $S = 1$. Solving (94) for $\cot(\alpha_1)$ yields $\cot(\alpha_1) = (\cos(\phi) + z^{1/(d-2)})/\sin(\phi)$. Similarly, in (94), we can rewrite $\sin(\alpha_1 - \phi)/\sin(\alpha_1)$ as $(\cos(\phi) + \sin(\phi)\cot(\alpha_1 - \phi))^{-1}$ and solve for $\cot(\alpha_1 - \phi) = -(\cos(\phi) + z^{-1/(d-2)})/\sin(\phi)$. Multiplying these yields

$$\Theta = \frac{-\cos^2(\phi) - 1 - \left(z^{1/(d-2)} + z^{-1/(d-2)}\right)\cos(\phi)}{\sin^2(\phi)},$$

which is maximized at $\phi = \pi/2$ with maximum value $\Theta = -1$. ∎

In Figure 11 we show α_1, α_d, and μ/η for $d = 5$ and various z. We omit the $z = 1$ case because $\alpha_1 = \phi/2$ and $\alpha_d = \phi/2 + \pi/2$ are constant and $\mu/\eta < -1$. For $z = -1/2$, we can see where the α_1 and α_d curves cross, yielding the nonhyperbolic, symmetric stationary point in Lemma 17. This point shows up in Figure 8 as a discontinuity in $\tilde{h}(G_1)$.

We also plotted, but do not include, the $d = 6$ case, which by Lemma 16 has no saddles when $0 < z$ and two saddles when $z < 0$. We find that the values of α_1, α_d, and μ/η corresponding to the $S = 1$ saddles are similar to the $d = 5$ case with the same z and the values corresponding to the $S = -1$ saddles are similar to the $d = 5$ case with $|z|$.

5.4.2. Saddles with $m > 1$. For $m > 1$ we are unable to solve (63) analytically for α_1, so we cannot conduct a full analysis. We did numerically find solutions to (63) as indicated in Table 2, and from them compute μ/η, but we cannot be certain that we captured all phenomena. From testing various combinations of (d, m, z), we observe the following:

- For $z > 0$, there are cases with $-1 < \Theta$, in contrast to the $m = 1$ case in Lemma 18. However Θ is never close to 0.
- For $z < 0$, $m > 1$, and d odd, when ϕ is sufficiently small there are 3 solutions to (63), rather than the one guaranteed by Table 2. The extra solutions correspond to the solutions when d is even, m is even, and $z < 0$, which only sometimes exist.
- Similarly, for $z < 0$, $m > 1$ and odd, and d even, when ϕ is sufficiently small there are 4 solutions rather than 2.
- We can observe directly from Lemma 13 that only $\Theta = 1$ can yield a nonhyperbolic stationary point. Lemma 17 shows this symmetric stationary point exists as a continuous extension of an $m = 1$ nonsymmetric saddle. We find that if $z < 0$ then this symmetric stationary point also exists as a continuous extension of nonsymmetric saddles for every $1 < m \leq d/2$. Thus for parameter configurations near those identified in Lemma 17 there are many bad saddles, especially taking into account the possible permutations of directions. We illustrate this situation in Figure 12.

5.4.3. Numerical validation of a swamp candidate. By Lemma 17, setting $(d, \phi, \lambda, z) = (6, \pi/8, 0, (95) \approx -0.19932)$ results in a nonhyperbolic saddle located at $\boldsymbol{\alpha} = (\phi/2 + \pi/4)\mathbf{1}$. Starting at $\boldsymbol{\alpha} = 1.0001(\phi/2 + \pi/4)\mathbf{1}$, the ALS algorithm has a transient swamp of length 7135, which we plot in Figure 13. For many intermediate iterations the error is very close to one and the change in error falls below 10^{-16} and so cannot be accurately computed. Plots of $\boldsymbol{\alpha}$

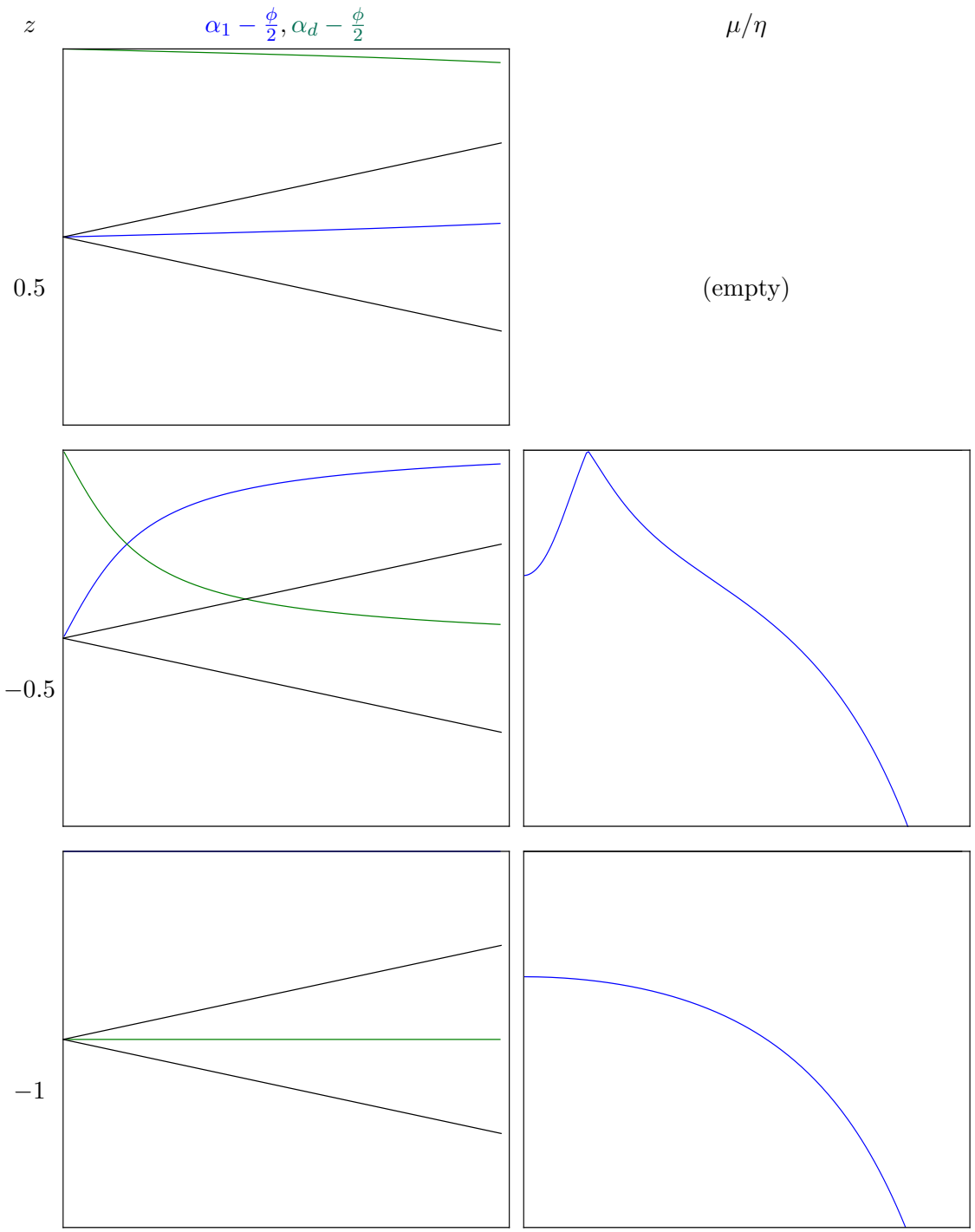

Figure 11. *Assessment of the saddle for $d = 5$ and $m = 1$ for various z. In each plot, the horizontal axis is the angle $\phi \in [0, \pi/2]$ defining the problem. The first column shows $\alpha_1 - \frac{\phi}{2}$ and $\alpha_d - \frac{\phi}{2}$, both in $(-\frac{\pi}{2}, \frac{\pi}{2}]$; the downward and upward slanted lines give the positions of the first and second terms in the target. The second column shows μ/η with vertical range $[-1, 0]$; when $\mu/\eta < -1$ the algorithm would escape in one or two steps.*

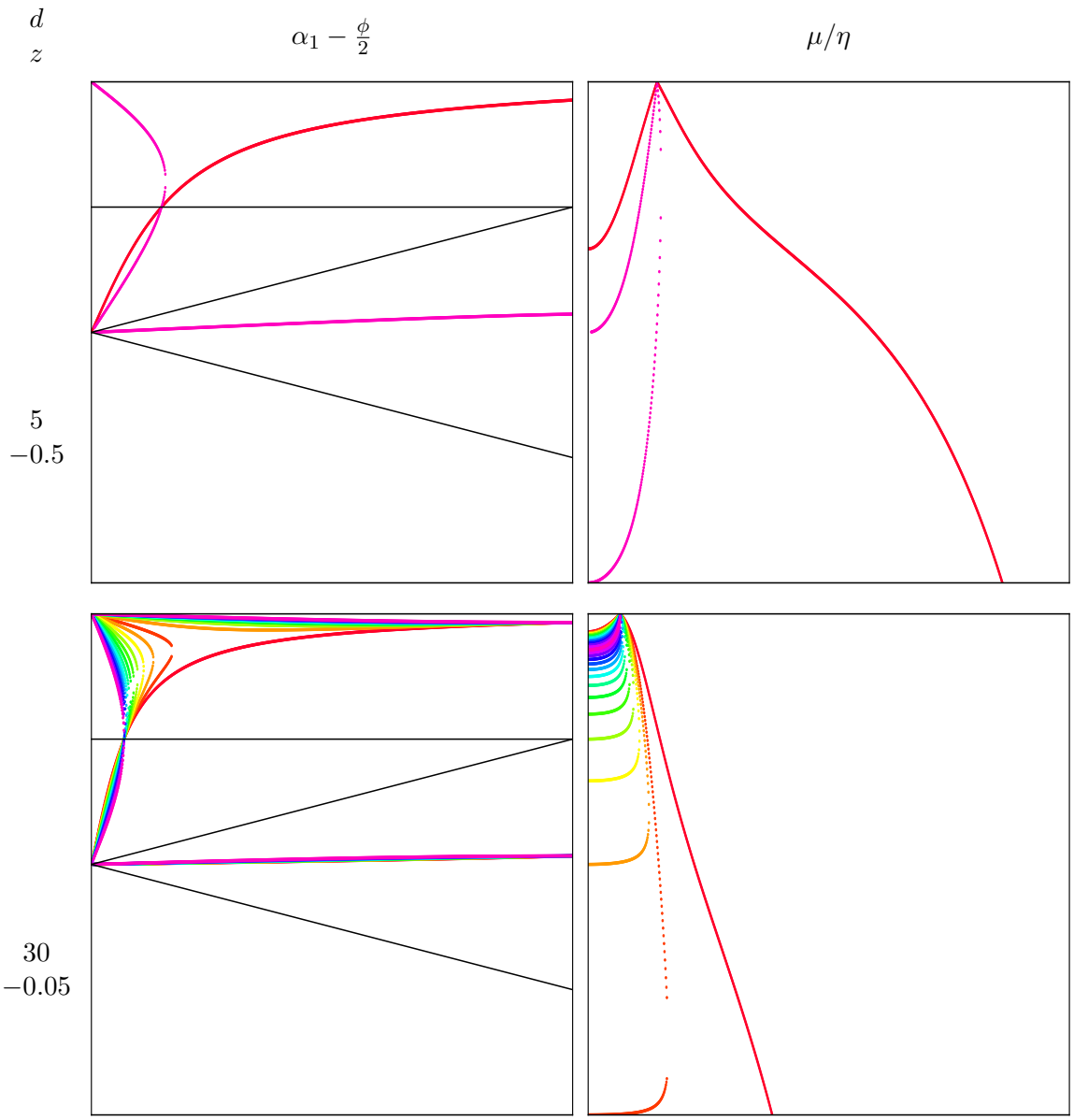

Figure 12. *Assessment of the saddles for* $(d, z; m) = (5, -0.5; 1, 2)$ *and* $(d, z; m) = (30, -0.05; 1, \ldots, 15)$. *In each plot, the horizontal axis is the angle* $\phi \in [0, \pi/2]$ *defining the problem. The first column shows* $\alpha_1 - \frac{\phi}{2}$ *in* $(-\frac{\pi}{2}, \frac{\pi}{2}]$ *colored by* m *in rainbow order; the downward and upward slanted lines give the positions of the first and second terms in the target and the horizontal line gives* $\alpha_1 = \phi/2 + \pi/4$. *The second column shows* μ/η *with vertical range* $[-1, 0]$, *colored by* m *in rainbow order. For small* ϕ, *a vertical line intersects multiple curves, indicating the presence of multiple saddles, each of which is really several saddles due to possible permutations of the directions.*

show gradual breaking of symmetry during most of the swamp, strong breaking of symmetry at the end of the swamp, and then restoration of symmetry during the final convergence phase.

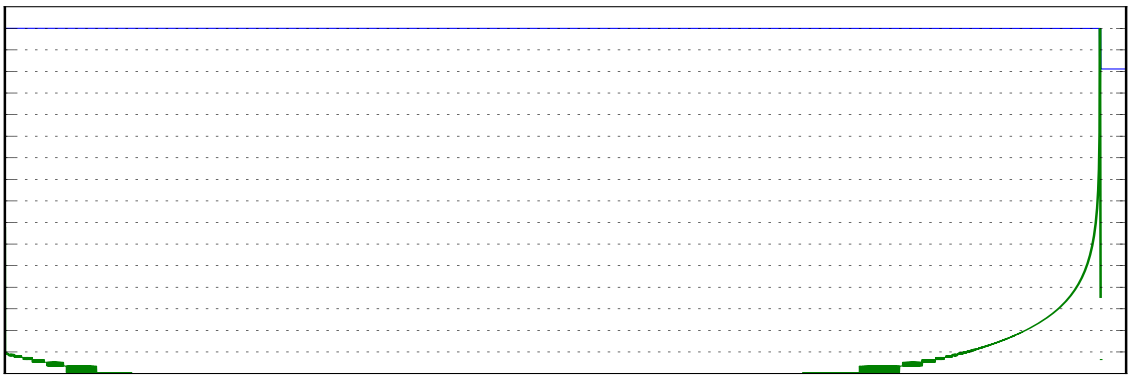

Figure 13. *Illustration of the transient swamp caused by passing near a nonhyperbolic saddle. The top (blue) curve is* $\log_{10}$ *of the error and the bottom (green) curve is* $\log_{10}$ *of the difference in error of consecutive iterations, with vertical axis* $[-16, 1]$*. The horizontal axis is the iteration number, in* $[0, 7400]$*.*

6. Analysis with partially symmetric T and G_1. In sections 4 and 5 we only considered symmetric T, which have $\boldsymbol{\phi} = \phi\mathbf{1}$. Analysis with general $\boldsymbol{\phi}$ seems beyond reach since there are too many parameters. We also suppose that we have discovered enough, if not all, of the important phenomena that occur when fitting a rank-2 tensor by a rank-1 tensor. In this section we briefly consider the case when T and G_1 are partially symmetric with $\phi_1 = \cdots = \phi_m \neq \phi_{m+1} = \cdots = \phi_d$ and $\alpha_1 = \cdots = \alpha_m \neq \alpha_{m+1} = \cdots = \alpha_d$. We set $z = 1$ and compare with the fully symmetric case in subsection 4.3. We confirm that the bifurcation phenomena with nonhyperbolic stationary points still occur and so are not specific to the symmetric case.

By inserting into the gradient (29), we see that $(\alpha_1, \alpha_d) = (\phi_1/2, \phi_d/2)$ is a stationary point. The eigenvalues of the Hessian at this stationary point can be computed by plugging into the definition (30), simplifying as in Lemma 12, and transforming as in Lemma 13. The eigenvalues are $2(n(\boldsymbol{\alpha}))^2(1+\lambda)^{-1}$ times

$$1 + \tan^2(\phi_1/2) \quad \text{with multiplicity } m-1, \tag{96}$$

$$1 + \tan^2(\phi_d/2) \quad \text{with multiplicity } d-m-1, \text{ and the eigenvalues of} \tag{97}$$

$$\begin{bmatrix} 1-(m-1)\tan^2(\phi_1/2) & \sqrt{m(d-m)}\tan(\phi_1/2)\tan(\phi_d/2) \\ \sqrt{m(d-m)}\tan(\phi_1/2)\tan(\phi_d/2) & 1-(d-m-1)\tan^2(\phi_d/2) \end{bmatrix}. \tag{98}$$

Since (96) and (97) are positive, this stationary point is stable transverse to the partial symmetry. The bifurcation value corresponding to ϕ_0 in (45) occurs when the determinant of (98) is zero, which is when

$$\phi_d = 2\arctan\left(\frac{1}{\sqrt{d-1}}\sqrt{\frac{1-(m-1)\tan^2(\phi_1/2)}{1-\frac{m}{d-1}+\tan^2(\phi_1/2)}}\right).$$

In Figure 14 we illustrate this relationship between ϕ_1 and ϕ_d.

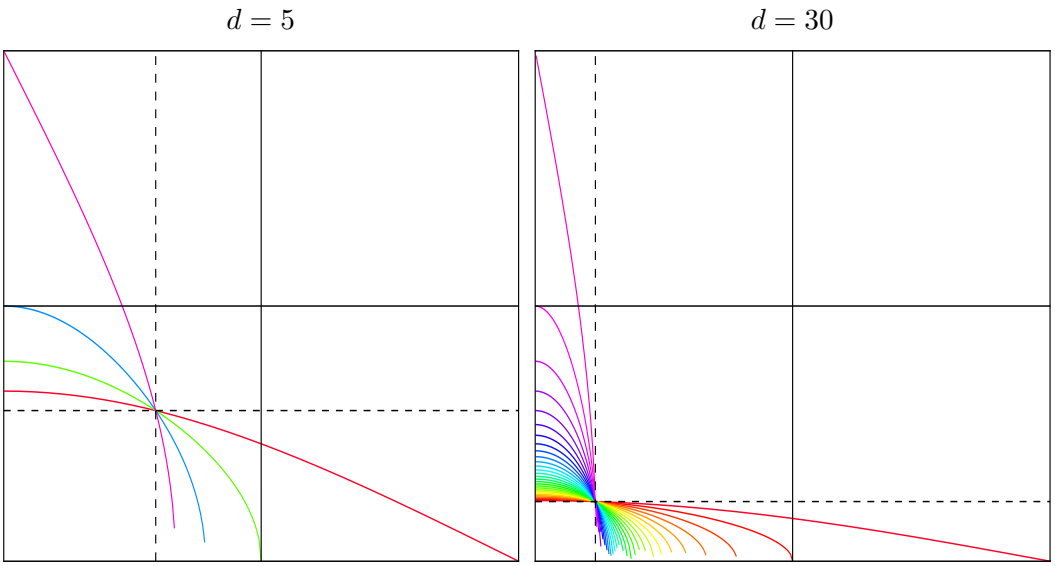

Figure 14. *Locations of the nonhyperbolic stationary point for partially symmetric T and G_1 for $d = 5$ with $m = 1, \ldots, 4$ and $d = 30$ with $m = 1, \ldots, 29$. In each plot, the horizontal axis is the angle $\phi_1 \in [0, \pi]$ with the symmetric bifurcation value ϕ_0 marked with a dashed line and $\pi/2$ with a solid line. Similarly, the vertical axis is ϕ_d. When either angle exceeds $\pi/2$ we can reflect it by $\phi \mapsto \pi - \phi$ and set $z = -1$.*

REFERENCES

[1] P. A. Absil, R. Mahony, and B. Andrews, *Convergence of the iterates of descent methods for analytic cost functions*, SIAM J. Optim., 16 (2005), pp. 531–547, https://doi.org/10.1137/040605266.

[2] E. Acar, D. M. Dunlavy, and T. G. Kolda, *A scalable optimization approach for fitting canonical tensor decompositions*, J. Chemometrics, 25 (2011), pp. 67–86, https://doi.org/10.1002/cem.1335.

[3] A. Ammar, F. Chinesta, and A. Falcó, *On the convergence of a greedy rank-one update algorithm for a class of linear systems*, Arch. Comput. Methods Eng., 17 (2010), pp. 473–486, https://doi.org/10.1007/s11831-010-9048-z.

[4] M. Bachmayr and W. Dahmen, *Adaptive near-optimal rank tensor approximation for high-dimensional operator equations*, Found. Comput. Math., 15 (2015), pp. 839–898, https://doi.org/10.1007/s10208-013-9187-3.

[5] M. Bachmayr and W. Dahmen, *Adaptive low-rank methods for problems on Sobolev spaces with error control in* L_2, ESAIM Math. Model. Numer. Anal., 50 (2016), pp. 1107–1136, https://doi.org/10.1051/m2an/2015071.

[6] M. Bachmayr, R. Schneider, and A. Uschmajew, *Tensor networks and hierarchical tensors for the solution of high-dimensional partial differential equations*, Found. Comput. Math., 16 (2016), pp. 1423–1472, https://doi.org/10.1007/s10208-016-9317-9.

[7] M. Balducci, B. Jones, and A. Doostan, *Orbit uncertainty propagation and sensitivity analysis with separated representations*, Celest. Mech. Dynam. Astronon., 129 (2017), pp. 105–136, https://doi.org/10.1007/s10569-017-9767-7.

[8] G. Beylkin, J. Garcke, and M. J. Mohlenkamp, *Multivariate regression and machine learning with sums of separable functions*, SIAM J. Sci. Comput., 31 (2009), pp. 1840–1857, https://doi.org/10.1137/070710524.

[9] G. Beylkin and M. J. Mohlenkamp, *Numerical operator calculus in higher dimensions*, Proc. Natl. Acad. Sci. USA, 99 (2002), pp. 10246–10251, https://doi.org/10.1073/pnas.112329799.

[10] G. Beylkin and M. J. Mohlenkamp, *Algorithms for numerical analysis in high dimensions*, SIAM J. Sci. Comput., 26 (2005), pp. 2133–2159, https://doi.org/10.1137/040604959.

[11] G. Beylkin, M. J. Mohlenkamp, and F. Pérez, *Approximating a wavefunction as an unconstrained sum of Slater determinants*, J. Math. Phys., 49 (2008), 032107, https://doi.org/10.1063/1.2873123.

[12] M. Billaud-Friess, A. Nouy, and O. Zahm, *A tensor approximation method based on ideal minimal residual formulations for the solution of high-dimensional problems*, ESAIM Math. Model. Numer. Anal., 48 (2014), pp. 1777–1806, https://doi.org/10.1051/m2an/2014019.

[13] K.-H. Boehm, A. A. Auer, and M. Espig, *Tensor representation techniques for full configuration interaction: A Fock space approach using the canonical product format*, J. Chem. Phys., 144 (2016), https://doi.org/10.1063/1.4953665.

[14] R. Botts, *Recovery and Analysis of Regulatory Networks from Expression Data Using Sums of Separable Functions*, Ph.D. thesis, Ohio University, Athens, OH, 2010, http://rave.ohiolink.edu/etdc/view?acc_num=ohiou1275926172.

[15] R. Bro, *PARAFAC. Tutorial and applications*, Chemometrics Intell. Lab. Syst., 38 (1997), pp. 149–171, https://doi.org/10.1016/S0169-7439(97)00032-4.

[16] R. Bro, *Multi-way Analysis in the Food Industry: Models, Algorithms, and Applications*, Ph.D. thesis, Universiteit van Amsterdam, Amsterdam, 1998.

[17] J. Buczyński and J. M. Landsberg, *Ranks of tensors and a generalization of secant varieties*, Linear Algebra Appl., 438 (2013), pp. 668–689, https://doi.org/10.1016/j.laa.2012.05.001.

[18] J. Buczyński and J. M. Landsberg, *On the third secant variety*, J. Algebraic Combin., 40 (2014), pp. 475–502, https://doi.org/10.1007/s10801-013-0495-0.

[19] H.-J. Bungartz and M. Griebel, *Sparse grids*, Acta Numer., 13 (2004), pp. 147–269, https://doi.org/10.1017/S0962492904000182.

[20] E. Cancès, V. Ehrlacher, and T. Lelièvre, *Greedy algorithms for high-dimensional eigenvalue problems*, Constr. Approx., 40 (2014), pp. 387–423, https://doi.org/10.1007/s00365-014-9266-y.

[21] J. D. Carroll and J. J. Chang, *Analysis of individual differences in multidimensional scaling via an N-way generalization of Eckart-Young decomposition*, Psychometrika, 35 (1970), pp. 283–320, https://doi.org/10.1007/BF02310791.

[22] B. Chen, S. He, Z. Li, and S. Zhang, *Maximum block improvement and polynomial optimization*, SIAM J. Optim., 22 (2012), pp. 87–107, https://doi.org/10.1137/110834524.

[23] M. Chevreuil, R. Lebrun, A. Nouy, and P. Rai, *A least-squares method for sparse low rank approximation of multivariate functions*, SIAM/ASA J. Uncertain. Quantif., 3 (2015), pp. 897–921, https://doi.org/10.1137/13091899X.

[24] S. R. Chinnamsetty, M. Espig, B. N. Khoromskij, W. Hackbusch, and H.-J. Flad, *Tensor product approximation with optimal rank in quantum chemistry*, J. Chem. Phys., 127 (2007), 084110, https://doi.org/10.1063/1.2761871.

[25] A. Cichocki, D. P. Mandic, A. H. Phan, C. F. Caiafa, G. Zhou, Q. Zhao, and L. De Lathauwer, *Tensor decompositions for signal processing applications*, IEEE Signal Process. Mag., 32 (2015), pp. 145–163, https://doi.org/10.1109/MSP.2013.2297439.

[26] P. Comon, X. Luciani, and A. L. F. de Almeida, *Tensor decompositions, alternating least squares and other tales*, J. Chemometrics, 23 (2009), pp. 393–405, https://doi.org/10.1002/cem.1236.

[27] P. Comon, J. M. F. ten Berge, L. De Lathauwer, and J. Castaing, *Generic and typical ranks of multi-way arrays*, Linear Algebra Appl., 430 (2009), pp. 2997–3007, https://doi.org/10.1016/j.laa.2009.01.014.

[28] T. L. D. Croft and T. N. Phillips, *Least-squares proper generalized decompositions for weakly coercive elliptic problems*, SIAM J. Sci. Comput., 39 (2017), pp. A1366–A1388, https://doi.org/10.1137/15M1049269.

[29] M. d'Avezac, R. Botts, M. J. Mohlenkamp, and A. Zunger, *Learning to predict physical properties using sums of separable functions*, SIAM J. Sci. Comput., 33 (2011), pp. 3381–3401, https://doi.org/10.1137/100805959.

[30] L. De Lathauwer, B. De Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1253–1278, https://doi.org/10.1137/S0895479896305696.

[31] L. De Lathauwer, B. De Moor, and J. Vandewalle, *On the best rank-1 and rank-*$(R_1, R_2, \cdots, R_N)$ *approximation of higher-order tensors*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 1324–1342, https://doi.org/10.1137/S0895479898346995.

[32] V. de Silva and L.-H. Lim, *Tensor rank and the ill-posedness of the best low-rank approximation problem*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1084–1127, https://doi.org/10.1137/06066518X.

[33] H. De Sterck, *A nonlinear GMRES optimization algorithm for canonical tensor decomposition*, SIAM J. Sci. Comput., 34 (2012), pp. A1351–A1379, https://doi.org/10.1137/110835530.

[34] H. De Sterck and M. Winlaw, *A nonlinearly preconditioned conjugate gradient algorithm for rank-R canonical tensor approximation*, Numer. Linear Algebra Appl., 22 (2015), pp. 410–432, https://doi.org/10.1002/nla.1963.

[35] I. Domanov and L. De Lathauwer, *On the uniqueness of the canonical polyadic decomposition of third-order tensors—Part* I: *Basic results and uniqueness of one factor matrix*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 855–875, https://doi.org/10.1137/120877234.

[36] I. Domanov and L. De Lathauwer, *On the uniqueness of the canonical polyadic decomposition of third-order tensors—Part* II: *Uniqueness of the overall decomposition*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 876–903, https://doi.org/10.1137/120877258.

[37] A. Doostan and G. Iaccarino, *A least-squares approximation of partial differential equations with high-dimensional random inputs*, J. Comput. Phys., 228 (2009), pp. 4332–4345, https://doi.org/10.1016/j.jcp.2009.03.006.

[38] M. Espig and W. Hackbusch, *A regularized Newton method for the efficient approximation of tensors represented in the canonical tensor format*, Numer. Math., 122 (2012), pp. 489–525, https://doi.org/10.1007/s00211-012-0465-9.

[39] M. Espig, W. Hackbusch, T. Rohwedder, and R. Schneider, *Variational calculus with sums of elementary tensors of fixed rank*, Numer. Math., 122 (2012), pp. 469–488, https://doi.org/10.1007/s00211-012-0464-x.

[40] N. Faber, R. Bro, and P. Hopke, *Recent developments in CANDECOMP/PARAFAC algorithms: A critical review*, Chemometrics Intell. Lab. Syst., 65 (2003), pp. 119–137, https://doi.org/10.1016/S0169-7439(02)00089-8.

[41] R. Fletcher and C. M. Reeves, *Function minimization by conjugate gradients*, Comput. J., 7 (1964), pp. 149–154, https://doi.org/10.1093/comjnl/7.2.149.

[42] S. Friedland, *On the generic and typical ranks of* 3*-tensors*, Linear Algebra Appl., 436 (2012), pp. 478–497, https://doi.org/10.1016/j.laa.2011.05.008.

[43] S. Friedland, *Best rank one approximation of real symmetric tensors can be chosen symmetric*, Front. Math. China, 8 (2013), pp. 19–40, https://doi.org/10.1007/s11464-012-0262-x.

[44] G. H. Golub and D. P. O'Leary, *Some history of the conjugate gradient and Lanczos algorithms:* 1948–1976, SIAM Rev., 31 (1989), pp. 50–102, https://doi.org/10.1137/1031003.

[45] X. Gong, *Dynamical Systems in Cell Division Cycle, Winnerless Competition Models, and Tensor Approximations*, Ph.D. thesis, Ohio University, Athens, Ohio, 2016, http://rave.ohiolink.edu/etdc/view?acc_num=ohiou1458303716.

[46] L. Grasedyck, *Hierarchical singular value decomposition of tensors*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 2029–2054, https://doi.org/10.1137/090764189.

[47] M. Griebel and J. Hamaekers, *Sparse grids for the Schrödinger equation*, ESAIM Math. Model. Numer. Anal., 41 (2007), pp. 215–247, https://doi.org/10.1051/m2an:2007015.

[48] M. Griebel and S. Knapek, *Optimized general sparse grid approximation spaces for operator equations*, Math. Comp., 78 (2009), pp. 2223–2257, https://doi.org/10.1090/S0025-5718-09-02248-0.

[49] W. Hackbusch, *Numerical tensor calculus*, Acta Numer., 23 (2014), pp. 651–742, https://doi.org/10.1017/S0962492914000087.

[50] W. Hackbusch and B. N. Khoromskij, *Low-rank Kronecker-product approximation to multi-dimensional nonlocal operators.* I. *Separable approximation of multi-variate functions*, Computing, 76 (2006), pp. 177–202, https://doi.org/10.1007/s00607-005-0144-0.

[51] W. Hackbusch and B. N. Khoromskij, *Low-rank Kronecker-product approximation to multi-dimensional nonlocal operators.* II. *HKT representation of certain operators*, Computing, 76 (2006), pp. 203–225, https://doi.org/10.1007/s00607-005-0145-z.

[52] R. A. Harshman, *Foundations of the PARAFAC procedure: Model and conditions for an "explanatory" multi-mode factor analysis*, in Working Papers in Phonetics 16, UCLA, Los Angeles, 1970, http://www.psychology.uwo.ca/faculty/harshman/wpppfac0.pdf.

[53] G. Hek, *Geometric singular perturbation theory in biological practice*, J. Math. Biol., 60 (2010), pp. 347–386, https://doi.org/10.1007/s00285-009-0266-7.

[54] C. J. Hillar and L.-H. Lim, *Most tensor problems are NP-hard*, J. ACM, 60 (2013), 45, https://doi.org/10.1145/2512329.

[55] M. W. Hirsch, S. Smale, and R. L. Devaney, *Differential Equations, Dynamical Systems, and an Introduction to Chaos*, 3rd ed., Academic Press, Waltham, MA, 2013, https://doi.org/10.1016/B978-0-12-382010-5.00001-4.

[56] F. L. Hitchcock, *The expression of a tensor or a polyadic as a sum of products*, J. Math. Phys., 6 (1927), pp. 164–189, https://doi.org/10.1002/sapm192761164.

[57] P. Hopke, P. Paatero, H. Jia, R. Ross, and R. Harshman, *Three-way (PARAFAC) factor analysis: Examination and comparison of alternative computational methods as applied to ill-conditioned data*, Chemometrics Intell. Lab. Syst., 43 (1998), pp. 25–42, https://doi.org/10.1016/S0169-7439(98)00077-X.

[58] A. S. Householder, *Unitary triangularization of a nonsymmetric matrix*, J. ACM, 5 (1958), pp. 339–342, https://doi.org/10.1145/320941.320947.

[59] M. Ishteva, P.-A. Absil, S. Van Huffel, and L. De Lathauwer, *Best low multilinear rank approximation of higher-order tensors, based on the Riemannian trust-region scheme*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 115–135, https://doi.org/10.1137/090764827.

[60] C. K. R. T. Jones, *Geometric singular perturbation theory*, in Dynamical Systems (Montecatini Terme, 1994), Lecture Notes in Math. 1609, Springer, Berlin, 1995, pp. 44–118, https://doi.org/10.1007/BFb0095239.

[61] S. Kaniel, *Estimates for some computational techniques in linear algebra*, Math. Comp., 20 (1966), pp. 369–378, https://doi.org/10.2307/2003590.

[62] V. A. Kazeev and E. E. Tyrtyshnikov, *The structure of the Hessian and an economical implementation of Newton's method in the problem of canonical approximation of tensors*, Comput. Math. Math. Phys., 50 (2010), pp. 929–945, https://doi.org/10.1134/S0965542510060011.

[63] B. N. Khoromskij, *Tensors-structured numerical methods in scientific computing: Survey on recent advances*, Chemometrics Intell. Lab. Syst., 110 (2012), pp. 1–19, https://doi.org/10.1016/j.chemolab.2011.09.001.

[64] B. N. Khoromskij and I. V. Oseledets, *QTT approximation of elliptic solution operators in higher dimensions*, Russian J. Numer. Anal. Math. Modelling, 26 (2011), pp. 303–322, https://doi.org/10.1515/RJNAMM.2011.017.

[65] B. N. Khoromskij and C. Schwab, *Tensor-structured Galerkin approximation of parametric and stochastic elliptic PDEs*, SIAM J. Sci. Comput., 33 (2011), pp. 364–385, https://doi.org/10.1137/100785715.

[66] E. Kofidis and P. A. Regalia, *On the best rank-1 approximation of higher-order supersymmetric tensors*, SIAM J. Matrix Anal. Appl., 23 (2002), pp. 863–884, https://doi.org/10.1137/S0895479801387413.

[67] T. G. Kolda, *Orthogonal tensor decompositions*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 243–255.

[68] T. G. Kolda and B. W. Bader, *Tensor decompositions and applications*, SIAM Rev., 51 (2009), pp. 455–500, https://doi.org/10.1137/07070111X.

[69] W. P. Krijnen, T. K. Dijkstra, and A. Stegeman, *On the non-existence of optimal solutions and the occurrence of "degeneracy" in the CANDECOMP/PARAFAC model*, Psychometrika, 73 (2008), pp. 431–439, https://doi.org/10.1007/s11336-008-9056-1.

[70] P. M. Kroonenberg and J. de Leeuw, *Principal component analysis of three-mode data by means of alternating least squares algorithms*, Psychometrika, 45 (1980), pp. 69–97, https://doi.org/10.1007/BF02293599.

[71] J. B. Kruskal, *Three-way arrays: Rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics*, Linear Algebra Appl., 18 (1977), pp. 95–138, https://doi.org/10.1016/0024-3795(77)90069-6.

[72] F. Lanzara, V. Maz'ya, and G. Schmidt, *On the fast computation of high dimensional volume potentials*, Math. Comp., 80 (2011), pp. 887–904, https://doi.org/10.1090/S0025-5718-2010-02425-1.

[73] F. Lanzara, V. Maz'ya, and G. Schmidt, *Fast cubature of volume potentials over rectangular domains by approximate approximations*, Appl. Comput. Harmon. Anal., 36 (2014), pp. 167–182, https://doi.org/10.1016/j.acha.2013.06.003.

[74] F. Lanzara and G. Schmidt, *On the computation of high-dimensional potentials of advection-diffusion operators*, Mathematika, 61 (2015), pp. 309–327, https://doi.org/10.1112/S0025579314000412.

[75] S. Leurgans and R. T. Ross, *Multilinear models: Applications in spectroscopy*, Statist. Sci., 7 (1992), pp. 289–319, http://www.jstor.org/stable/2246065.

[76] S. E. Leurgans, R. T. Ross, and R. B. Abel, *A decomposition for three-way arrays*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1064–1083, https://doi.org/10.1137/0614071.

[77] N. Li, *Variants of ALS on Tensor Decompositions and Applications*, Ph.D. thesis, Clarkson University, Potsdam, NY, 2013.

[78] N. Li, S. Kindermann, and C. Navasca, *Some convergence results on the regularized alternating least-squares method for tensor decomposition*, Linear Algebra Appl., 438 (2013), pp. 796–812, https://doi.org/10.1016/j.laa.2011.12.002.

[79] B. C. Mitchell and D. S. Burdick, *Slowly converging PARAFAC sequences: Swamps and two-factor degeneracies*, J. Chemometrics, 8 (1994), pp. 155–168, https://doi.org/10.1002/cem.1180080207.

[80] M. J. Mohlenkamp, *A center-of-mass principle for the multiparticle Schrödinger equation*, J. Math. Phys., 51 (2010), 022112, https://doi.org/10.1063/1.3290747.

[81] M. J. Mohlenkamp, *Capturing the interelectron cusp using a geminal layer on an unconstrained sum of Slater determinants*, SIAM J. Appl. Math., 72 (2012), pp. 1742–1771, https://doi.org/10.1137/110823900.

[82] M. J. Mohlenkamp, *Function space requirements for the single-electron functions within the multiparticle Schrödinger equation*, J. Math. Phys., 54 (2013), 062105, https://doi.org/10.1063/1.4811396.

[83] M. J. Mohlenkamp, *Musings on multilinear fitting*, Linear Algebra Appl., 438 (2013), pp. 834–852, https://doi.org/10.1016/j.laa.2011.04.019.

[84] C. Navasca, L. D. Lathauwer, and S. Kinderman, *Swamp reducing technique for tensor decomposition*, in 16th European Signal Processing Conference, EUSIPCO 2008, 2008, European Association for Signal Processing, Lausanne, Switzerland, http://ieeexplore.ieee.org/document/7080724/.

[85] A. Nouy, *A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations*, Comput. Methods Appl. Mech. Engrg., 199 (2010), pp. 1603–1626, https://doi.org/10.1016/j.cma.2010.01.009.

[86] A. Nouy, *Proper generalized decompositions and separated representations for the numerical solution of high dimensional stochastic problems*, Arch. Comput. Methods Eng., 17 (2010), pp. 403–434, https://doi.org/10.1007/s11831-010-9054-1.

[87] I. Oseledets, *DMRG approach to fast linear algebra in the TT-format*, Comput. Methods Appl. Math., 11 (2011), pp. 382–393, https://doi.org/10.2478/cmam-2011-0021.

[88] I. V. Oseledets, *Tensor-train decomposition*, SIAM J. Sci. Comput., 33 (2011), pp. 2295–2317, https://doi.org/10.1137/090752286.

[89] I. V. Oseledets and S. V. Dolgov, *Solution of linear systems and matrix inversion in the TT-format*, SIAM J. Sci. Comput., 34 (2012), pp. A2718–A2739, https://doi.org/10.1137/110833142.

[90] I. V. Oseledets and E. E. Tyrtyshnikov, *Breaking the curse of dimensionality, or how to use SVD in many dimensions*, SIAM J. Sci. Comput., 31 (2009), pp. 3744–3759, https://doi.org/10.1137/090748330.

[91] P. Paatero, *The multilinear engine—a table-driven, least squares program for solving multilinear problems, including the n-way parallel factor analysis model*, J. Comput. Graph. Statist., 8 (1999), pp. 854–888, https://doi.org/10.2307/1390831.

[92] P. Paatero, *Construction and analysis of degenerate PARAFAC models*, J. Chemometrics, 14 (2000), pp. 285–299, https://doi.org/10.1002/1099-128X(200005/06)14:3⟨285::AID-CEM584⟩3.3.CO;2-T.

[93] P. Paatero, P. Hopke, X. Song, and Z. Ramadan, *Understanding and controlling rotations in factor analytic models*, Chemometrics Intell. Lab. Syst., 60 (2002), pp. 253–264, https://doi.org/10.1016/S0169-7439(01)00200-3.

[94] P. Paatero and P. K. Hopke, *Rotational tools for factor analytic models*, J. Chemometrics, 23 (2009), pp. 91–100, https://doi.org/10.1002/cem.1197.

[95] A.-H. Phan, P. Tichavský, and A. Cichocki, *Low complexity damped Gauss–Newton algorithms for CANDECOMP/PARAFAC*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 126–147, https://doi.org/10.1137/100808034.

[96] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes*, Cambridge University Press, Cambridge, 1986.

[97] E. Pruliere, F. Chinesta, and A. Ammar, *On the deterministic solution of multidimensional parametric models using the proper generalized decomposition*, Math. Comput. Simulation, 81 (2010), pp. 791–810, https://doi.org/10.1016/j.matcom.2010.07.015.

[98] M. Rajih, P. Comon, and R. A. Harshman, *Enhanced line search: A novel method to accelerate PARAFAC*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 1128–1147, https://doi.org/10.1137/06065577.

[99] M. V. Rakhuba and I. V. Oseledets, *Fast multidimensional convolution in low-rank tensor formats via cross approximation*, SIAM J. Sci. Comput., 37 (2015), pp. A565–A582, https://doi.org/10.1137/140958529.

[100] W. S. Rayens and M. C. Mitchell, *Two-factor degeneracies and a stabilization of PARAFAC*, Chemometrics Intell. Lab. Syst., 38 (1997), pp. 173–181.

[101] M. J. Reynolds, A. Doostan, and G. Beylkin, *Randomized alternating least squares for canonical tensor decompositions: Application to a PDE with random data*, SIAM J. Sci. Comput., 38 (2016), pp. A2634–A2664, https://doi.org/10.1137/15M1042802.

[102] R. Schneider and A. Uschmajew, *Convergence results for projected line-search methods on varieties of low-rank matrices via Łojasiewicz inequality*, SIAM J. Optim., 25 (2015), pp. 622–646, https://doi.org/10.1137/140957822.

[103] J. R. Shewchuk, *An introduction to the conjugate gradient method without the agonizing pain*, Technical report, Carnegie Mellon University, Pittsburgh, PA, 1994, http://www.cs.cmu.edu/~quake-papers/painless-conjugate-gradient.pdf.

[104] L. Sorber, M. Van Barel, and L. De Lathauwer, *Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in rank-$(L_r, L_r, 1)$ terms, and a new generalization*, SIAM J. Optim., 23 (2013), pp. 695–720, https://doi.org/10.1137/120868323.

[105] L. Sorber, M. Van Barel, and L. De lathauwer, *Structured data fusion*, IEEE J. Sel. Top. Signal Process., 9 (2015), pp. 586–600, https://doi.org/10.1109/JSTSP.2015.2400415.

[106] M. Sørensen, L. De Lathauwer, P. Comon, S. Icart, and L. Deneire, *Canonical polyadic decomposition with a columnwise orthonormal factor matrix*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1190–1213, https://doi.org/10.1137/110830034.

[107] A. Stegeman, *Degeneracy in CANDECOMP/PARAFAC explained for $p \times p \times 2$ arrays of rank $p + 1$ or higher*, Psychometrika, 71 (2006), pp. 483–501, https://doi.org/10.1007/s11336-004-1266-6.

[108] A. Stegeman, *Degeneracy in CANDECOMP/PARAFAC and INDSCAL explained for several three-sliced arrays with a two-valued typical rank*, Psychometrika, 72 (2007), pp. 601–619, https://doi.org/10.1007/s11336-007-9022-3.

[109] A. Stegeman, *Low-rank approximation of generic $p \times q \times 2$ arrays and diverging components in the Candecomp/Parafac model*, SIAM J. Matrix Anal. Appl., 30 (2008), pp. 988–1007, https://doi.org/10.1137/050644677.

[110] A. Stegeman, *On uniqueness conditions for CANDECOMP/PARAFAC and INDSCAL with full column rank in one mode*, Linear Algebra Appl., 431 (2009), pp. 211–227, https://doi.org/10.1016/j.laa.2009.02.025.

[111] A. Stegeman, *On uniqueness of the nth order tensor decomposition into rank-1 terms with linear independence in one mode*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 2498–2516, https://doi.org/10.1137/090779632.

[112] A. Stegeman, *On uniqueness of the canonical tensor decomposition with some form of symmetry*, SIAM J. Matrix Anal. Appl., 32 (2011), pp. 561–583, https://doi.org/10.1137/100814615.

[113] A. Stegeman, *Candecomp/Parafac: From diverging components to a decomposition in block terms*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 291–316, https://doi.org/10.1137/110825327.

[114] A. Stegeman, *A three-way Jordan canonical form as limit of low-rank tensor approximations*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 624–650, https://doi.org/10.1137/120875806.

[115] A. Stegeman, *Finding the limit of diverging components in three-way CANDECOMP/PARAFAC —a demonstration of its practical merits*, Comput. Statist. Data Anal., 75 (2014), pp. 203–216, https://doi.org/10.1016/j.csda.2014.02.010.

[116] A. Stegeman and A. L. F. de Almeida, *Uniqueness conditions for constrained three-way factor decompositions with linearly dependent loadings*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 1469–1490, https://doi.org/10.1137/080743354.

[117] A. Stegeman and L. De Lathauwer, *A method to avoid diverging components in the Candecomp/Parafac model for generic $I \times J \times 2$ arrays*, SIAM J. Matrix Anal. Appl., 30 (2009), pp. 1614–1638, https://doi.org/10.1137/070692121.

[118] A. Stegeman and T. T. T. Lam, *Improved uniqueness conditions for canonical tensor decompositions with linearly dependent loadings*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 1250–1271, https://doi.org/10.1137/110847275.

[119] A. Stegeman and N. D. Sidiropoulos, *On Kruskal's uniqueness condition for the CANDECOMP/PARAFAC decomposition*, Linear Algebra Appl., 420 (2007), pp. 540–552, https://doi.org/10.1016/j.laa.2006.08.010.

[120] A. Stegeman and J. M. F. Ten Berge, *Kruskal's condition for uniqueness in CANDECOMP/PARAFAC when ranks and k-ranks coincide*, Comput. Statist. Data Anal., 50 (2006), pp. 210–220, https://doi.org/10.1016/j.csda.2004.07.015.

[121] A. Stegeman, J. M. F. ten Berge, and L. De Lathauwer, *Sufficient conditions for uniqueness in CANDECOMP/PARAFAC and INDSCAL with random component matrices*, Psychometrika, 71 (2006), pp. 219–229, https://doi.org/10.1007/11336-006-1278-2.

[122] Y. Sun and M. Kumar, *Uncertainty propagation in orbital mechanics via tensor decomposition*, Celestial Mech. Dynam. Astronom., 124 (2016), pp. 269–294, https://doi.org/10.1007/s10569-015-9662-z.

[123] J. M. F. ten Berge, *Least squares optimization in multivariate analysis*, M&T Ser. 25, D.S.W.O. Press, Leiden, The Netherlands, 1993.

[124] P. Tichavsky, A.-H. Phan, and A. Cichocki, *Partitioned alternating least squares technique for canonical polyadic tensor decomposition*, IEEE Signal Process. Lett., 23 (2016), pp. 993–997, https://doi.org/10.1109/LSP.2016.2577383.

[125] G. Tomasi and R. Bro, *PARAFAC and missing values*, Chemometrics Intell. Lab. Syst., 75 (2005), pp. 163–180, https://doi.org/10.1016/j.chemolab.2004.07.003.

[126] G. Tomasi and R. Bro, *A comparison of algorithms for fitting the PARAFAC model*, Comput. Statist. Data Anal., 50 (2006), pp. 1700–1734, https://doi.org/10.1016/j.csda.2004.11.013.

[127] A. Uschmajew, *Well-posedness of convex maximization problems on Stiefel manifolds and orthogonal tensor product approximations*, Numer. Math., 115 (2010), pp. 309–331, https://doi.org/10.1007/s00211-009-0276-9.

[128] A. Uschmajew, *Local convergence of the alternating least squares algorithm for canonical tensor approximation*, SIAM J. Matrix Anal. Appl., 33 (2012), pp. 639–652, https://doi.org/10.1137/110843587.

[129] A. Uschmajew, *A new convergence proof for the higher-order power method and generalizations*, Pac. J. Optim., 11 (2015), pp. 309–321.

[130] A. Validi, *Low-rank separated representation surrogates of high-dimensional stochastic functions: Application in Bayesian inference*, J. Comput. Phys., 260 (2014), pp. 37–53, https://doi.org/10.1016/j.jcp.2013.12.024.

[131] L. Wang and M. T. Chu, *On the global convergence of the alternating least squares method for rank-one approximation to generic tensors*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 1058–1072, https://doi.org/10.1137/130938207.

[132] L. Wang, M. T. Chu, and B. Yu, *Orthogonal low rank tensor approximation: Alternating least squares method and its global convergence*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 1–19, https://doi.org/10.1137/130943133.

[133] H. Yserentant, *On the regularity of the electronic Schrödinger equation in Hilbert spaces of mixed derivatives*, Numer. Math., 98 (2004), pp. 731–759, https://doi.org/10.1007/s00211-003-0498-1.

[134] H. Yserentant, *Sparse grid spaces for the numerical solution of the electronic Schrödinger equation*, Numer. Math., 101 (2005), pp. 381–389, https://doi.org/10.1007/s00211-005-0581-x.

[135] H. Yserentant, *The hyperbolic cross space approximation of electronic wavefunctions*, Numer. Math., 105 (2007), pp. 659–690, https://doi.org/10.1007/s00211-006-0038-x.

[136] H. Yserentant, *Regularity and Approximability of Electronic Wave Functions*, Lecture Notes in Math. 2000, Springer, Berlin, 2010, https://doi.org/10.1007/978-3-642-12248-4.

[137] T. Zhang and G. H. Golub, *Rank-one approximation to high order tensors*, SIAM J. Matrix Anal. Appl., 23 (2001), pp. 534–550, https://doi.org/10.1137/S0895479899352045.