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, algorithms to do so are not robust and regularly exhibit unpleasantly interesting 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 non-hyperbolic minima and saddles, which would be difficult for algorithms to handle. These features occur at relatively small but non-zero 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 Multilinear algebra, tensor products; 37N30 Dynamical systems in numerical analysis; 65K10 Optimization and variational techniques

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

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

or without indexes as

\[ T \approx G = \sum_{l=1}^{r} \bigotimes_{i=1}^{d} G_{li}. \]

This sum-of-separable format for $G$ is known variously as the Canonical format, the Canonical Polyadic (CP) format, the CANDECOMP/PARAFAC (CP) format, or a separated representation (e.g. [7, 8, 19, 21, 44, 47, 50, 51, 61, 98]). Such an approximation has applications in data analysis and signal processing (e.g. [6, 13, 19, 22, 47, 61, 68, 69]), numerical methods in high dimensions (e.g. [4, 5, 7–9, 11, 45, 46, 57, 65–67, 78, 79, 90, 114, 122]), uncertainty quantification (e.g. [32, 59, 79, 90]), and others (e.g. [6, 25]). Alternative tensor formats exist and work well for some applications (e.g. [41, 58, 80–83, 92]), but the sum-of-separable format is the most natural and appears to be the only format that can be used in other applications, such as the multiparticle Schrödinger equation in which antisymmetry is essential (e.g. [9, 11, 73–75]).

Funding: This material is based upon work supported by the National Science Foundation under Grant 1418787.

†Department of Mathematics, Ohio University, xg345709@ohio.edu
‡Department of Mathematics, Ohio University, mohlenka@ohio.edu
§Department of Mathematics, Ohio University, youngt@ohio.edu
There has been steady, but slow, progress in understanding aspects of the approximation problem \( (2) \) such as uniqueness (e.g. [23,30,31,64,86,87,102–104,108,110–113]), maximal and generic rank (e.g. [15, 24, 37]), limits (e.g. [16,27,62–99–101,105–107,109,119]), and approximability of function classes (e.g. [17,21,42,43,45,46,124–127]). 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 \( d > 2 \).

Numerical algorithms to solve \( (2) \) have been developed based on Alternating Least-Squares (ALS) (e.g. [7,8,20,23,51,63,69–71,76,77,85,93,94,115,120,121]), Newton’s method (e.g. [33,56, 88,117]), nonlinear conjugate gradient (e.g. [2,29,34,84]) line search (e.g. [23,91]), and others (e.g. [28, 53, 84, 97, 116]); see the discussion and comparisons in [35, 51, 118]. Unfortunately, these algorithms are prone to exhibit unpleasantly interesting behavior, such as long periods in which the iteration reduces the error by miniscule amounts, which are known informally as “swamps”. Work has been done trying to understand and alleviate swamps (e.g. [23,23,70–72,77,85,91,93]), but they remain a crucial impediment to the use of methods based on \( (2) \). In particular, for numerical methods in high dimensions based on [7, 8], many approximations like \( (2) \) are needed and the failure of a few of them can spoil the whole calculation. Indeed, the development of alternative tensor formats has largely been motivated by the lack of a robust algorithm to solve \( (2) \).

Our ultimate goal is to produce robust algorithms for solving \( (2) \) so that it can be used more effectively and more widely. Given the unsatisfying 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 \( \{G_l\} \), 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 the algorithms, which are mostly based on gradients. In particular, we would like 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. Knowledge of how these aspects change with \( T \) may help us understand which tensors are hard to approximate, and may suggest different algorithms for different types of \( T \). By tuning these parameters so that \( T \) has particularly bad swamps, we can gain an understanding of the enemy and obtain hard test cases on which to try out algorithms.

Analysis with general \( T \) and \( G \) is out of reach, so we consider a model problem. The simplest case of \( (2) \) is when \( T \) itself is separable. In this case our understanding is complete and standard algorithms can find \( G \) (which equals \( T \)) exactly. The next simplest case is when \( T \) has rank two. By normalizing and applying a tensor product of unitary matrices, any such (real) tensor can be put in the standard form

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

where \( |z| \leq 1, \phi_i \in [0, \pi/2] \), and the leading scalar ensures \( \|T\| = 1 \). The tensor product of unitary matrices commutes with the operations in algorithms such as ALS, so it suffices to
consider \( T \) in (3). The simplest case for \( G \), in the coordinate system induced by (3), is to choose \( r = 1 \) and let

\[
G_1 = a \bigotimes_{i=1}^{d} \begin{bmatrix}
\cos(\alpha_i) \\
\sin(\alpha_i)
\end{bmatrix}.
\]

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. [26, 26, 60, 121, 123, 128]) and some “greedy” algorithms are based on repeatedly fitting with \( r = 1 \) (e.g. [3, 10, 18, 78, 79, 90]). Once the approximation of \( T \) in (3) by \( G_1 \) is understood, one should consider \( G_2 \) 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 \( T \) in (3) by \( G_1 \), \( G_2 \), and \( G_3 \), and conduct the analysis for \( G_1 \) in (4). 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.

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 $\phi$ and unstable for larger $\phi$.

- We show visualizations for various $d$ and $z$ illustrating the dependence on $\phi$ and $\alpha$ 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 $\phi$ 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 $\phi$ 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.

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 [40, Chapter 4].

- We describe all global maxima.
- We determine when non-symmetric 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, non-hyperbolic 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 non-hyperbolic 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 [35, 51, 118].

2. The worst features, in the form of non-hyperbolic stationary points, occur at discrete, non-zero angles. In the literature, swamps are associated with small angle (called ill-
conditioning or degeneracy) (see e.g. [23, 71, 72, 77, 91, 93]), 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. [14, 47, 51, 69, 91, 118]). We believe that our analysis using transverse stability explains such behavior. This analysis supports the use of extrapolation methods (e.g. [14, 28, 47, 51, 69, 91, 118]) and methods based on nonlinear conjugate gradient (e.g. [2, 29, 34, 84]).

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 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 non-standard 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 \( \vec{1}_k \) denote the (column) vector of size \( k \) with entries 1, \( \vec{0}_k \) denote the (column) vector of size \( k \) with entries 0, and \( I_k \) denote the identity matrix of size \( k \times k \); if the subscript \( k \) is missing it is assumed to be \( d \). Let \( e_j \) denote the vector with 1 in entry \( j \) and otherwise 0. For \( 1 \leq q \leq p \), let the matrix

\[
U_{pq} = I_p - 2 \frac{(e_q - \vec{1}_p/\sqrt{p})(e_q - \vec{1}_p/\sqrt{p})^*}{(e_q - \vec{1}_p/\sqrt{p})^*(e_q - \vec{1}_p/\sqrt{p})}
\]

be the Householder reflector [52] that takes \( \vec{1}_p \) to \( \sqrt{p} e_q \); it is unitary and is its own conjugate transpose and inverse.

2.1. Dynamical Systems Background. Suppose we have a differentiable function \( f(x) \) defined on some connected domain \( D \). By the Gradient Flow of \( f \) we mean the flow induced by the differential equation

\[
\dot{x}(t) = -\nabla f(x(t)).
\]

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 \( f \) \[49\]. Since we have assumed that \( f \) is differentiable, the critical points of \( f \) are the points \( x \) where \( \nabla f(x) = 0 \), which are precisely the *equilibrium* or *stationary* points of the flow defined by (6).

The stability of the flow at an equilibrium \( x \) is often a starting point for understanding a dynamical system. It is well-known that the local stability at \( x \) 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 \( f(x) \). Since the Hessian is symmetric, it has only real eigenvalues. Without loss of generality, suppose \( f \) has a critical point at \( 0 \) and \( f(0) = 0 \). Letting \( H \) denote the Hessian of \( f \) at \( 0 \), we approximately have

\[
(7) \quad f(x) = \frac{1}{2} x^* H x \quad \text{and} \quad \nabla f(x) = H x .
\]

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

Away from stationary points, the flow moves like \( t \| \nabla f(x) \| \), 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 when and then 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 \( f(x) \) with parameters evolving under the gradient flow defined by \( \dot{x}(t) = -\nabla f(x(t)) \). We can consider two stability questions about the flow:

1. Given some value \( x \) and a nearby value \( x_1 \), will the flows from \( x \) and \( x_1 \) locally converge or diverge?
2. Given that the flow from \( x \) should stay in some invariant set, due for example to symmetry, if we start from a nearby value \( x_1 \) not in this set, will the flow from \( x_1 \) converge to or diverge from this set?

In both cases the strategy is:

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

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

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

For an illustration of this effect, see Figure 1. Although the sign of $\mu(x)$ is sufficient to determine if the flow is (un)stable at $x$, the magnitude of $\mu(x)$ does not allow meaningful comparisons of how (un)stable the flow is at two different points. To allow such comparisons we will use $h(x)$.

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 general position. We consider the gradient-descent minimization algorithm with line-search: From an initial point $x$, compute the gradient $\nabla f(x)$, find $t$ to minimize $f(x - t\nabla f(x))$, set $x_{\text{new}} = x - t\nabla f(x)$, 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 $\mu$ 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 $\mu$, but by the ratio $\mu/\eta$, where $\eta$ is the maximum eigenvalue of $H$.
- At a saddle, the progress is also governed by the ratio $\mu/\eta$.

The behavior of gradient descent at a minimum is well-known (see e.g. [89,96]), and is a prime motivation for using conjugate-gradient algorithms (see e.g. [36,39,89]). 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 $\|\nabla f(x)\|$. 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 $\|\nabla f(x)\|/\eta$, where $\eta$ 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. [36] and ridges in [47, p. 31]).

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

$$f(x - t\nabla f(x)) = \frac{1}{2}(x - tHx)^tH(x - tHx) = \frac{t^2}{2}x^tH^3x - tx^tH^2x + x^tHx.$$  \hspace{1cm} (9)

If $x^tH^3x < 0$, then (9) has no minimum with respect to $t$, so the gradient-descent minimization algorithm would move to infinity. If $x^tH^3x > 0$, then the minimum of (9) occurs when

$$t = \frac{x^tH^2x}{x^tH^3x}, \quad \text{which yields} \quad x_{\text{new}} = x - \frac{x^tH^2x}{x^tH^3x}Hx.$$  \hspace{1cm} (10)

At a maximum all eigenvalues of $H$ are negative, so $x^tH^3x < 0$ 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 $x^tH^3x > 0$ so the algorithm will take a finite step toward the stationary point. At a saddle the sign of $x^tH^3x$ depends on $x$.

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

$$H = \begin{bmatrix} \mu & 0 \\ 0 & \eta \end{bmatrix}, \quad \text{which yields} \quad x_{\text{new}} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - \frac{\mu^2x_1^2 + \eta^2x_2^2}{\mu^2x_1^2 + \eta^2x_2^2} \begin{bmatrix} \mu x_1 \\ \eta x_2 \end{bmatrix} = \frac{1}{1 + \left(\frac{\mu}{\eta}\right)^2\left(\frac{x_1}{x_2}\right)^2} \begin{bmatrix} 1 - \frac{\mu}{\eta} \\ \left(\frac{x_1}{x_2}\right)^2 - \frac{\mu}{\eta} \end{bmatrix} x_2.$$  \hspace{1cm} (11)

The second update is

$$(x_{\text{new}})_{\text{new}} = \frac{\left(1 - \frac{\mu}{\eta}\right)^2}{1 + \frac{\mu}{\eta}\left(\frac{\mu x_1}{\eta x_2}\right)^2 + \left(\frac{\mu x_1}{\eta x_2}\right)^{-2}} + \left(\frac{\mu}{\eta}\right)^2 x_1 \begin{bmatrix} 1 \\ x_2 \end{bmatrix},$$  \hspace{1cm} (12)

which is on the same ray as $x$. As we continue to update, every two updates multiplies by the scalar in (13).

For a minimum, we have $0 < \mu < \eta$ and the scalar in (13) is in $(0, 1)$, with smaller values giving faster convergence. With $0 < \mu < \eta$ fixed, the scalar is maximized when $|\mu x_1/\eta x_2| = 1$. Taking the square root to give the net contraction factor per iteration then gives the scalar

$$\frac{1 - \mu/\eta}{1 + \mu/\eta}. \hspace{1cm} (14)$$
**Figure 2.** 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 $\pm \mu/\eta$.

**Figure 3.** 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.

**Figure 4.** 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 $x_2 = \pm \delta/\eta$. 
Thus convergence is fastest as $\mu/\eta \to 1^-$ and slowest as $\mu/\eta \to 0^+$. See Figure 2 for an illustration.

For a saddle we have $\mu < 0 < \eta$ and there are three possible behaviors. First, if $x^* H^3 x < 0$ then the algorithm moves far from the stationary point in a single step. Second, if $x^* H^3 x > 0$ but $x_{new}^* H^3 x_{new} < 0$ then the algorithm moves far from the stationary point in the second step. Third, if $x^* H^3 x > 0$ and $x_{new}^* H^3 x_{new} > 0$ the iteration proceeds according to (13). In terms of ratios, these conditions are

\[
\begin{align*}
(15) \quad & x^* H^3 x > 0 \iff \mu^3 x_1^2 + \eta x_2^2 > 0 \iff \left| \frac{x_1}{x_2} \right| < \left( \frac{\mu}{\eta} \right)^{-3/2} \\
(16) \quad & x_{new}^* H^3 x_{new} > 0 \iff -\left( \frac{\mu}{\eta} \right)^{-2} \left( \frac{x_1}{x_2} \right)^{-1} \left( \frac{\mu}{\eta} \right)^{-3/2} \iff \left( \frac{\mu}{\eta} \right)^{-1/2} < \left| \frac{x_1}{x_2} \right|,
\end{align*}
\]

which can be combined into

\[
(17) \quad \left| \frac{x_1}{x_2} \right| \in \left( \left( \frac{\mu}{\eta} \right)^{-1/2}, \left( \frac{\mu}{\eta} \right)^{-3/2} \right).
\]

Note that if $\mu/\eta \leq -1$ 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 $x$. Assuming (17) holds, the scalar in (13) 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 (14); notice that now $\mu/\eta < 0$. Thus divergence is fastest as $\mu/\eta \to -1^+$ and slowest as $\mu/\eta \to 0^-$. See Figure 3 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 $\exp((14)^k$ in $k$ steps. To understand the difference, we can either compare $\exp(-\mu)$ to (14) or compare $\mu$ to

\[
(18) \quad -\ln((14)) = -\ln \left( \frac{1 - \frac{\mu}{\eta}}{1 + \frac{\mu}{\eta}} \right) = \ln \left( 1 + \frac{\mu}{\eta} \right) - \ln \left( 1 - \frac{\mu}{\eta} \right) = \frac{2\mu}{\eta} + O \left( \left( \frac{\mu}{\eta} \right)^3 \right).
\]

For more than two eigenvalues the iteration is not clean like (13). We will let $\mu$ be the minimum eigenvalue of $H$ and $\eta$ be the maximum eigenvalue. For a minimum, the ratio $\eta/\mu$ is the condition number of $H$ and the contraction factor (14) has been established as a bound (see e.g. [96] which uses [55]). For a saddle, $\mu/\eta$ 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 $\mu/\eta$ 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 $x$ with negative gradient of norm $\delta$ pointing in the $e_1$ direction and diagonal Hessian with eigenvalue 0 corresponding to $e_1$ and eigenvalue $\eta$ corresponding to $e_2$. (Thus $\eta$ is the eigenvalue of the transverse Hessian.) Thus we have

\[
f(x) = x^* \begin{bmatrix} -\delta & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{2} x^* \begin{bmatrix} 0 & 0 \\ 0 & \eta \end{bmatrix} x = -\delta x_1 + \frac{1}{2} \eta x_2^2.
\]
The gradient and Hessian of \( f \) are

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

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

\[
f(x - t \nabla f(x)) = f \left( \frac{x_1 + tx}{x_2 - t\eta x_2} \right) = \frac{t^2}{2} \eta x_2^2 - t(\delta^2 + \eta^2 x_2^2) - \delta x_1 + \frac{1}{2} \eta x_2^2.
\]

If \( \eta < 0 \), then (21) 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 (21) occurs when

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

\[
x_{\text{new}} = x - \frac{\delta^2 + \eta^2 x_2^2}{\eta^2 x_2^2} \begin{bmatrix} -\delta \\ \eta x_2 \end{bmatrix} = \begin{bmatrix} x_1 + \frac{\delta}{\eta} \left(1 + \left(\frac{\delta}{\eta x_2}\right)^2 x_2^{-2}\right) \\ -\delta \left(\frac{\delta}{\eta x_2}\right)^2 x_2^{-1} \end{bmatrix}
\]

and

\[
(x_{\text{new}})_{\text{new}} = \begin{bmatrix} x_1 + \frac{\delta}{\eta} \left(2 + \left(\frac{\delta}{\eta x_2}\right)^2 + \left(\frac{\delta}{\eta x_2}\right)^{-2}\right) \\ x_2 \end{bmatrix}.
\]

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

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

which attains its minimum value of \( 2\delta/\eta \) when \( |x_2| = \delta/\eta \). In subsection 2.1, the flow moved like \( t \| \nabla f(x) \| \), but here we find progress like \( k^2 \| \nabla f(x) \| /\eta \). See Figure 4 for an illustration.

If the transverse Hessian has eigenvalues other than \( \eta \) then the iteration will not be so clean. If all its eigenvalues are positive we can take \( \eta \) 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 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 \( x_1 \) and \( x_2 \) with \( f(x_1) < f(x_2) \). Given only this information, we would prefer the point \( x_1 \) to the point \( x_2 \). Now suppose we also know \( \| \nabla f(x_1) \| < \| \nabla f(x_2) \| \), so that the gradient flow from \( x_1 \) is slower than the gradient flow from \( x_2 \). 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 \( f(x) \) and \( \| \nabla f(x) \| \). This quantity is interpreted as a local estimate of the time it would take for the flow from \( x \) to reach a root (which is also a minimum) of \( f \). Although we have not found this method in the literature, we would be surprised if it is new.
Table 1

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

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

$$s(x) = \frac{f(x)}{\|\nabla f(x)\|_2} \frac{1}{\|\nabla f(x)\|_2} = \frac{f(x)}{\|\nabla f(x)\|_2}.$$

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

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

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

For a differentiable $f$ with $x_*$ in the interior of $D$, the flow will not reach $x_*$ in finite time, so the estimate (26) 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 $x_*$ such that $f(x_*) = 0$. 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 $s(x)$ is measuring near the minimum, it is providing useful qualitative information.

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

$$|f(x) - f(x_*)| \leq C\|\nabla f(x)\|_2^{\theta}.$$

Under certain assumptions on an iterative, gradient-based method for minimizing $f$, its iterates $x_k$ will approach a local minimizer $x_*$ and the error will satisfy

$$\|x_* - x_k\| = \begin{cases} O(q^k) & \text{if } \theta = 2 \text{ (for some } 0 < q < 1), \\ O(k^{-(\theta-1)/(2-\theta)}) & \text{if } 1 < \theta < 2. \end{cases}$$
Thus linear convergence is only established when (28) holds with $\theta = 2$ (for that local minimizer $x_\ast$). For general discussion of this theory see [1, 95] and for its use in tensor approximations see [120, 121]. Since we desire to approach the global minimizer, we set $f(x_\ast) = 0$. Since we desire to approach with at least linear convergence, we set $\theta = 2$. With these choices, (28) is equivalent to $s(x) \leq C$ holding uniformly in a neighborhood of $x_\ast$. Pointwise, $s(x)$ estimates how well an iterative method starting at $x$ can be expected to converge to a $x_\ast$ with $f(x_\ast) = 0$. Small values indicate that (28) is easily satisfied and linear convergence is expected. Large values indicate that (28) either requires a large $C$, which can be expected to slow convergence by increasing $q$, or that it requires a smaller $\theta$, which would mean slower than linear convergence.

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

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 $G_1$ in (4). 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

$$F = \sum_{m=1}^{R} f_m \bigotimes_{i=1}^{d} F_i^m \quad \text{and} \quad G = \sum_{l=1}^{r} g_l \bigotimes_{i=1}^{d} G_i^l,$$

then

$$\langle F, G \rangle = \sum_{m=1}^{R} \sum_{l=1}^{r} f_m g_l \prod_{i=1}^{d} \langle F_i^m, G_i^l \rangle.$$

We measure the quality of the approximation using regularized least-squares error (in the sense used in [8, 76]), which reduces to

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

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

\[ n(\vec{\alpha}) = \left( \bigotimes_{i=1}^{d} \begin{bmatrix} \cos(\alpha_i) \\ \sin(\alpha_i) \end{bmatrix}, T \right) = \frac{\prod_{i=1}^{d} \cos(\alpha_i) + z \prod_{i=1}^{d} \cos(\alpha_i - \phi_i)}{\left(1 + 2z \prod_{i=1}^{d} \cos(\phi_i) + z^2\right)^{1/2}}, \]

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

\[ n_j(\vec{\alpha}) = \frac{\partial}{\partial \alpha_j} n(\vec{\alpha}) = -\sin(\alpha_j) \prod_{i=1, i\neq j}^{d} \cos(\alpha_i) - z \sin(\alpha_j - \phi_j) \prod_{i=1, i\neq j}^{d} \cos(\alpha_i - \phi_i) \]

\[ \left(1 + 2z \prod_{i=1}^{d} \cos(\phi_i) + z^2\right)^{1/2}. \]

When \( j \neq k \), its second partial derivatives are

\[ n_{jk}(\vec{\alpha}) = \frac{\partial}{\partial \alpha_k} n_j(\vec{\alpha}) = \frac{\sin(\alpha_j) \sin(\alpha_k) \prod_{i=1, i\neq j,k}^{d} \cos(\alpha_i) + z \sin(\alpha_j - \phi_j) \sin(\alpha_k - \phi_k) \prod_{i=1, i\neq j,k}^{d} \cos(\alpha_i - \phi_i)}{\left(1 + 2z \prod_{i=1}^{d} \cos(\phi_i) + z^2\right)^{1/2}} \]

and when \( j = k \) they are

\[ n_{jj}(\vec{\alpha}) = \frac{\partial}{\partial \alpha_j} n_j(\vec{\alpha}) = -n(\vec{\alpha}). \]

### 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. [48,54]). 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 \( 1 \times 1 \) 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

\[ [1 + \lambda] [a] = [n(\vec{\alpha})]. \]
The coefficient can be determined by inverting the matrix to obtain

\[
\begin{bmatrix}
a \\
\end{bmatrix} = \begin{bmatrix} 1 + \lambda \\
\end{bmatrix}^{-1} \begin{bmatrix} n(\tilde{\alpha}) \\
\end{bmatrix} = \begin{bmatrix} n(\tilde{\alpha}) \\
\end{bmatrix} / (1 + \lambda). 
\]

**Remark 1.** The Alternating Least Squares 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

\[
E_\lambda(G_1) = \|T\|^2 - (\|G_1\|^2 + \lambda a^2) = 1 - (\|G_1\|^2 + \lambda a^2). 
\]

Evaluating directly and then using (40), we have

\[
\|G_1\|^2 + \lambda a^2 = a^2(1 + \lambda) = n^2(\tilde{\alpha}) / (1 + \lambda). 
\]

Therefore, combining (41) and (42), we obtain that

\[
E_\lambda(G_1) = 1 - n^2(\tilde{\alpha}) / (1 + \lambda). 
\]

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

\[
\frac{\partial}{\partial \alpha_j} E_\lambda(G_1) = \frac{\partial}{\partial \alpha_k} E_\lambda(G_1) = -2 \frac{n(\tilde{\alpha}) n_j(\tilde{\alpha})}{1 + \lambda} \quad \text{and} \\
\frac{\partial^2}{\partial \alpha_j \partial \alpha_k} E_\lambda(G_1) = \frac{\partial^2}{\partial \alpha_k \partial \alpha_j} E_\lambda(G_1) = -2 \frac{(n_k(\tilde{\alpha}) n_j(\tilde{\alpha}) + n(\tilde{\alpha}) n_{jk}(\tilde{\alpha}))}{1 + \lambda}. 
\]

4. Analysis with Symmetric \(T\) and \(G_1\). In this section we assume \(\tilde{\phi} = \phi \tilde{\alpha}\) and \(\tilde{\alpha} = \alpha \tilde{\alpha}\), 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 \(\tilde{\alpha} = \alpha \tilde{\alpha}\). 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 \(\tilde{\alpha} = \alpha \tilde{\alpha}\), 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.
4.1. Preliminaries. The auxiliary functions in subsection 3.1 become

\[ n(\vec{\alpha}) = \frac{\cos^d(\alpha) + z \cos^d(\alpha - \phi)}{(1 + 2z \cos^d(\phi) + z^2)^{1/2}}, \]

(46)

\[ n_j(\vec{\alpha}) = \frac{-\sin(\alpha) \cos^{d-1}(\alpha) - z \sin(\alpha - \phi) \cos^{d-1}(\alpha - \phi)}{(1 + 2z \cos^d(\phi) + z^2)^{1/2}}, \]

(47)

for \( j \neq k \)

\[ n_{jk}(\vec{\alpha}) = \frac{\sin^2(\alpha) \cos^{d-2}(\alpha) + z \sin^2(\alpha - \phi) \cos^{d-2}(\alpha - \phi)}{(1 + 2z \cos^d(\phi) + z^2)^{1/2}}, \]

(48)

\[ n_{jj}(\vec{\alpha}) = -(46). \]

The first and second derivatives of \( E_\lambda \) can be computed as

\[ \frac{d}{d\alpha} E_\lambda(G_1) = \tilde{\Gamma}_1^d \nabla E_\lambda(G_1) = \sum_{j=1}^{d} (44) = \frac{d(-2n(\vec{\alpha})n_j(\vec{\alpha}))}{1 + \lambda} \]

(50)

\[ \frac{d^2}{d\alpha^2} E_\lambda(G_1) = \tilde{\Gamma}_1^d H \tilde{\Gamma}_d = \sum_{j=1}^{d} \sum_{k=1}^{d} (45) = \frac{-2 \left( d^2 n_j^2(\vec{\alpha}) + d(d-1)n(\vec{\alpha})n_{jk}(\vec{\alpha}) - dn^2(\vec{\alpha}) \right)}{1 + \lambda}. \]

(51)

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

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

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

(52)

**Proof.** If \( n(\vec{\alpha}) = 0 \) then \( E_\lambda(G_1) = 1 - n^2(\vec{\alpha})/(1 + \lambda) = 1 \), which is a global maximum. Setting \( n(\vec{\alpha}) \) in (46) to 0 and manipulating yields

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

(53)

which requires that \( z < 0 \) or \( d \) is odd (or both) so that we can compute \((-z)^{-1/d}\). Continuing to solve for \( \alpha \) yields (52).

If \( z > 0 \) and \( d \) is even, then \( n(\vec{\alpha}) > 0 \) for all \( \alpha \) so we cannot locate the global maximum this way. To find other maxima or minima within the set \( \vec{\alpha} = \alpha \tilde{\Gamma} \), which may turn out to be saddles without this constraint, we set \( \frac{d}{d\alpha} E_\lambda(G_1) = 0 \) by setting \( n_j(\vec{\alpha}) = 0 \). If \( \phi = \pi/2 \) then \( \alpha = 0 \) and \( \alpha = \pi/2 \) solve \( n_j(\vec{\alpha}) = 0 \); they are minima since (51) yields \( \frac{d^2}{d\alpha^2} E_\lambda(G_1) = \frac{2dn^2(\vec{\alpha})}{1 + \lambda} > 0 \). Otherwise, we have to solve

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

(54)

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 $\vec{\alpha} = \alpha \vec{1}$, the transverse Hessian $H^\perp(\vec{\alpha})$ has the single eigenvalue

$$
2(\cos^d(\alpha) + z \cos^d(\alpha - \phi)) (\cos^d-2(\alpha) + z \cos^d-2(\alpha - \phi))
$$

with multiplicity $d - 1$.

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

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

Conjugating with the Householder reflector $U_d$ from (5) yields

$$
U_d^* H(\vec{\alpha}) U_d = (H_{11} - H_{12})I_d + dH_{12} e_1 e_1^* = 
\begin{bmatrix}
H_{11} + (d - 1)H_{12} & 0 \\
0 & (H_{11} - H_{12})I_{d-1}
\end{bmatrix}.
$$

Deleting the first row and column leaves $H^\perp(\vec{\alpha}) = (H_{11} - H_{12})I_{d-1}$, which has a single eigenvalue $(H_{11} - H_{12})$ with multiplicity $d - 1$. To compute this value, we start with (45) and insert (46), (47), (48), and (49), to obtain

$$
-2 \frac{2(47)^2 - (46)^2}{1 + \lambda} - \frac{2(47)^2 + (46)(48)}{1 + \lambda} = \frac{2(46)((46) + (48))}{1 + \lambda} = (55).
$$

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

(59) $d$ is odd, $z < 0$, and $\alpha \in [\alpha_+(d), \alpha_+(d - 2)]$;

(60) $d$ is odd, $0 < z$, and $\alpha \in [\alpha_+(d - 2), \alpha_+(d)]$; and

(61) $d$ is even, $z < 0$, and $\alpha \in [\alpha_-(d - 2), \alpha_-(d)] \cup [\alpha_+(d), \alpha_+(d - 2)]$;

(62) where $\alpha_\pm(k) = \arctan \left( \frac{\pm(-z)^{-1/2} - \cos(\phi)}{\sin(\phi)} \right)$.

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

Proof. If $d$ is even and $0 < z$ then (55) is always positive and so the flow is stable. Otherwise (55) changes sign when either of its factors do. Since (55) is positive at $\alpha = \pi/2$, its signs on the intervals are determined. When $d$ is odd, we can set each factor to zero and solve, to obtain (59) and (60). When $d$ is even and $z < 0$, we have the values from (59) and a second interval from the sign ambiguity.

Note that the first factor in (55) is $n(\vec{\alpha})$, 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. . . . 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

\[
(63) \quad (46) \iff n(\vec{\phi}/2) = \frac{(1 + z) \cos^d(\phi/2)}{(2 + 2z \cos^d(\phi))^{1/2}},
\]
\[
(64) \quad (47) \iff n_j(\vec{\phi}/2) = \frac{-(1 - z) \sin(\phi/2) \cos^{d-1}(\phi/2)}{(2 + 2z \cos^d(\phi))^{1/2}}, \quad \text{and}
\]
\[
(65) \quad (48) \iff n_{jk}(\vec{\phi}/2) = \frac{(1 + z) \sin^2(\phi/2) \cos^{d-2}(\phi/2)}{(2 + 2z \cos^d(\phi))^{1/2}},
\]

so \( n_j(\vec{\phi}/2) = 0 \) when \( z = 1 \) and \( n(\vec{\phi}/2) = n_{jk}(\vec{\phi}/2) = 0 \) 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

\[
(66) \quad \phi_0 = 2 \arcsin(d^{-1/2}) = 2 \arctan\left((d - 1)^{-1/2}\right),
\]

with a local minimum at \( \phi/2 \) when \( 0 < \phi < \phi_0 \) and a local maximum when \( \phi_0 < \phi \leq \pi/2 \).

**Proof.** Inserting (63) and (64) in (50) yields

\[
(67) \quad \frac{d^2}{d\alpha^2} E_\lambda(G_1) \bigg|_{\alpha = \phi/2} =
\]
\[
(1 + z)^2 \frac{-d \cos^{d-2}(\phi/2) \left(d \sin^2(\phi/2) - 1\right)}{(1 + \lambda)(1 + z \cos d(\phi))} + (1 - z)^2 \frac{-d^2 \sin^2(\phi/2) \cos^{2d-2}(\phi/2)}{(1 + \lambda)(1 + z \cos d(\phi))}.
\]

For \( z = -1 \), the first term in (67) drops out, and the result is strictly negative for \( 0 < \phi \leq \pi/2 \). 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 (67) drops out, and the result is zero at (66). For \( 0 < \phi < \phi_0 \), (67) 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 (55) is zero at \( \alpha = \phi/2 \). Thus this maximum in the symmetric set is non-hyperbolic in the full space.

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

\[
(68) \quad \frac{4 \cos^{2d-2}(\phi/2)}{(1 + \lambda)(1 + \cos^d(\phi))}.
\]

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 (57), is also \((67)/d\). Taking the ratio yields

\[
\frac{(67)/d}{(68)} = \frac{1 - d \sin^2(\phi/2)}{\cos^d(\phi/2)}.
\]

For \( \phi \) much larger than \( \phi_0 \), we will have \((69) < -1\) 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 \((69) \to 1\) and the analysis in subsection 2.3.1 shows an algorithm will converge rapidly to the minimum. Since \((69) = 0\) at \( \phi_0 \), 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 (26) to \( \tilde{s}(G) \in [0, 1] \) via

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

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 \( \mu \), which will also play the role of \( \eta \) in (27). For visualization purposes we will map \( v(\cdot) \in [0, \infty] \) from (27) to \( \tilde{v}(G) \in [0, 1] \) via

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

When \( \mu(G) < 0 \) then \( \tilde{v}(G) \) is undefined and will be plotted in white.

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 (8) to \([0, 1]\) and use

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

Values \( \tilde{h}(G) \in [0, 1/2) \) indicate stability and values \( \tilde{h}(G) \in (1/2, 1] \) indicate instability. When \( \|\nabla E_\lambda(G)\|_2 \gg |\mu| \), then \( \tilde{h}(G) \approx 1/2 \), indicating that the (in)stability would not have as much effect.

In Figures 5 to 7 we show example \( E_\lambda(G_1) \), \( \tilde{s}(G_1) \), \( \tilde{v}(G) \), and \( \tilde{h}(G_1) \) for \( d = 5, d = 6, \) and \( d = 30 \) with \( \lambda = 0 \). From our earlier analysis and the plots, we observe the following:
Figure 5. Error $E_0(G_1)$, 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. The horizontal axis is the angle $\phi \in [0, \pi/2]$ defining the problem, the vertical axis is the centered angle $\alpha - \phi/2 \in [-\pi/2, \pi/2]$. The downward and upward slanted lines give the positions of the first and second terms in the target. In the estimated algorithm time landscape, the contour $\mu = 2$ is overlaid in black. In the stability landscape the contour at $1/2$ is highlighted in black.
Figure 6. Landscapes for $d = 6$ and $\lambda = 0$ in the same format as Figure 5.
Figure 7. Landscapes for $d = 30$ and $\lambda = 0$ in the same format as Figure 5. For $z = -0.5$, the unstable region is poorly resolved by the plotting program.
• When $\phi$ exceeds $\pi/2$ the case $(d, z, \phi)$ maps to $(d, (-1)^d z, \pi - \phi)$. In the plots, for even $d$ one takes the plot, rotates it by $\pi$, shifts it up by $\pi/2$, and appends it to itself on the right. For odd $d$ one rotates the plot by $\pi$, 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 $\phi$.

• When $z = 1$, the gradient is small in a large neighborhood of the minima when $\phi \approx \phi_0$ from (66). This region corresponds to values of the ratio (69) 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 $\phi$ 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 $\vec{\alpha} = \alpha \vec{1}$ 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.

• As suggested by the special case (68), the eigenvalue $\mu$ 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 $z > 0$, $\phi$ is small, and $\alpha$ is near the global minimum. In some situations, such as the local maximum when $z = -1$ and $\alpha = \phi/2$, $\mu$ is small (in this case 0), which improves the situation.

• For $-1 < z < 0$, when $\phi$ 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 $\phi$ 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.

5. Analysis with Symmetric $T$ and Asymmetric $G$. In this section we assume $\vec{\phi} = \phi \vec{1}$ 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 (44) outside of the symmetric set. This section is based in part on the dissertation [40, 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 non-global 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 [38], [12, Chapter 7], and [20, Corollary 4.2]. Our results in subsection 5.3 are slightly stronger in that they exclude local minima and local (non-global) 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 $\vec{\alpha} = \alpha \vec{1}$ in Lemma 2.

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

\begin{align}
\alpha_j &= \pi/2 = \phi - \alpha_k \quad \text{for some } j \neq k \\
\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).
\end{align}

Proof. If $n(\vec{\alpha}) = 0$ then $E_\lambda(G_1) = 1 - n^2(\vec{\alpha})/(1 + \lambda) = 1$, which is a global maximum. Since $n(\vec{\alpha})$ in (34) is a sum of two products, there are two types of solutions to $n(\vec{\alpha}) = 0$.

If one product is zero then both must be zero. Therefore a factor in each must be zero, which yields (73). The values for $\alpha_i$ for $k \neq i \neq j$ are unconstrained.

If neither product is zero, then we can manipulate $n(\vec{\alpha}) = 0$ to isolate $\alpha_d$ 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 $\alpha_d$ yields (74).

5.2. Existence of Non-Symmetric stationary Points. To obtain the stationary points of the gradient flow, we set

\[ \frac{\partial}{\partial \alpha_j} E_\lambda(G_1) = -\frac{2n(\vec{\alpha})n_j(\vec{\alpha})}{1 + \lambda} = 0. \]

If $n(\vec{\alpha}) = 0$ then the point is a global maximum, as discussed in subsection 5.1; it may be symmetric or non-symmetric. We therefore assume $n(\vec{\alpha}) \neq 0$ and study when $n_j(\vec{\alpha}) = 0$ for all $j$. First, in Theorem 9, we show that non-symmetric 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 $i$, $n(\vec{\alpha}) = 0$, $n_i(\vec{\alpha}) = 0$ for all $i$, and $\alpha_1 \neq \alpha_k$ for some $k$, then, up to permutation of the directions,

\begin{align}
\alpha_1 &= \cdots = \alpha_m \neq \alpha_{m+1} = \alpha_{m+2} = \cdots = \alpha_d \quad \text{for some } m \in \{1, \ldots, \lfloor d/2 \rfloor\} \text{ and} \\
\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].
\end{align}

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

Proof. By (35), we know that

\[ 0 = \| T \| n_j(\vec{\alpha}) = -\sin(\alpha_j) \prod_{i \neq j} \cos(\alpha_i) - z \sin(\alpha_j - \phi) \prod_{i \neq j} \cos(\alpha_i - \phi). \]  

Select \( k \) such that \( \alpha_1 \neq \alpha_k \), multiply the \( n_k \) version of (79) by \( \sin(\alpha_1 - \phi) \cos(\alpha_k - \phi) \), multiply the \( n_1 \) version of (79) by \( \sin(\alpha_k - \phi) \cos(\alpha_1 - \phi) \), and subtract to obtain

\[ 0 = \left( \prod_{i \neq 1, \neq k} \cos(\alpha_i) \right) \left( \begin{array}{c} \sin(\alpha_1) \cos(\alpha_k) \sin(\alpha_k - \phi) \cos(\alpha_1 - \phi) \\ -\sin(\alpha_k) \cos(\alpha_1) \sin(\alpha_1 - \phi) \cos(\alpha_k - \phi) \end{array} \right). \]  

Direct expansion using sum-of-angles formulas shows that (80) is the same as

\[ 0 = \sin(\delta) \sin(\alpha_k - \alpha_1) \cos(\alpha_k + \alpha_1 - \phi) \prod_{i \neq 1, \neq k} \cos(\alpha_i). \]  

By our assumption \( \phi \in [0, \pi/2] \) we know \( \sin(\delta) \neq 0 \) 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 (81) requires \( \cos(\alpha_k + \alpha_1 - \phi) = 0 \), which implies

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

If there exists \( p \neq k \) such that \( \alpha_p \neq \alpha_1 \) then we can subtract the corresponding equations (82) 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 \( \alpha_1, \ldots, \alpha_d \).

For convenience, we permute them so that

\[ \alpha_1 = \cdots = \alpha_m \neq \alpha_{m+1} = \alpha_{m+2} = \cdots = \alpha_d \quad \text{for some } m \in \{1, \ldots, [d/2]\}. \]  

Under the additional assumption \( \alpha_i \neq \pi/2 \) for all \( i \), we have thus proven (77) and by applying (82) have proven (78).

Now suppose \( \alpha_p = \pi/2 \) for exactly one \( p \); by permuting we may assume \( p = 1 \). Then (81) requires \( \cos(\alpha_k + \pi/2 - \phi) = 0 \), which implies \( \alpha_k = \phi \) for all \( k \neq 1 \). Thus (77) and (78) hold in this case as well.

Finally suppose \( \alpha_p = \alpha_q = \pi/2 \) for some \( p \neq q \). Then (79) reduces to

\[ 0 = \| T \| n_j(\vec{\alpha}) = z \sin(\alpha_j - \phi) \prod_{i \neq j} \cos(\alpha_i - \phi) \]  

for all \( j \). In order for (84) 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 \( n(\vec{\alpha}) = 0 \), which contradicts our assumption.
The existence of non-symmetric 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 $\alpha_1$ with $\alpha_d$.

<table>
<thead>
<tr>
<th>$m$ odd</th>
<th>$m$ even</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-1 \leq z &lt; 0$</td>
<td>$0 &lt; z \leq 1$</td>
</tr>
<tr>
<td>$d$ odd</td>
<td>at least 1</td>
</tr>
<tr>
<td>$d$ even</td>
<td>at least 2</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Lemma 10.** If $z \neq 0$, $\phi \in (0, \pi/2]$, $\alpha_i \in (-\pi/2, \pi/2]$ for all $i$, $n(\bar{\alpha}) \neq 0$, (77), and (78), then $n_i(\bar{\alpha}) = 0$ for all $i$ if and only if

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

**Proof.** Due to (77), we need only consider $n_1(\bar{\alpha})$ and $n_d(\bar{\alpha})$. Plugging (83) into (79) for $j = 1$ and $j = d$ yields

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

and

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

From (78) we have

$$\cos(\alpha_d - \phi) = -\sin(\alpha_1),$$

$$\sin(\alpha_d - \phi) = -\cos(\alpha_1),$$

and

$$\cos(\alpha_d) = -\sin(\alpha_1 - \phi),$$

$$\sin(\alpha_d) = -\cos(\alpha_1 - \phi),$$

so we can eliminate $\alpha_d$ and obtain

$$0 = \sin(\alpha_1) \sin(\alpha_1 - \phi) \left( \cos^{m-1}(\alpha_1) \sin^{d-m-1}(\alpha_1 - \phi) + z \cos^{m-1}(\alpha_1 - \phi) \sin^{d-m-1}(\alpha_1) \right)$$

and

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

The condition (85) implies both (90) and (91), so one direction of the theorem 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 (90) or (91) imply (85). 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 (78) then implies $\alpha_1 = \alpha_d$, which violates (77).

**Theorem 11.** If $z \neq 0$, $\phi \in (0, \pi/2]$, $\alpha_i \in (-\pi/2, \pi/2]$ for all $i$, $n(\bar{\alpha}) \neq 0$, (77), and (78), then solutions to $n_i(\bar{\alpha}) = 0$ for all $i$ exist according to the cases in Table 2. For the case of $d$ even, $m$ even, and $z < 0$, solutions exist for

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

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

---

<table>
<thead>
<tr>
<th>$d$ odd</th>
<th>at least 1</th>
<th>at least 1</th>
<th>at least 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d$ even</td>
<td>at least 2</td>
<td>do not exist</td>
<td>sometimes exist</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>at least 2</td>
</tr>
</tbody>
</table>

**Table 2**

The existence of non-symmetric 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 $\alpha_1$ with $\alpha_d$. 

<table>
<thead>
<tr>
<th>$m$ odd</th>
<th>$m$ even</th>
</tr>
</thead>
<tbody>
<tr>
<td>$-1 \leq z &lt; 0$</td>
<td>$0 &lt; z \leq 1$</td>
</tr>
</tbody>
</table>
Proof. By Lemma 10 we need only determine whether or not (85) has solutions. Let \( f(\alpha_1) \) denote the right-hand-side of (85), 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

\[
\begin{align*}
(93) \quad f(-\pi/2) &= z \sin^{m-1}(\phi)(-1)^{d-2}, \\
(94) \quad f(\phi) &= z \sin^{d-m-1}(\phi), \quad \text{and} \quad f(\pi/2) = z \sin^{m-1}(\phi).
\end{align*}
\]

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 non-negative, 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 where \( f(\alpha_1) > 0 \). We first consider the case \( \phi = \pi/2 \), for which

\[
(95) \quad f(\alpha_1) = (-1)^{d-m-1}\cos^{d-2}(\alpha_1) + z \sin^{d-2}(\alpha_1).
\]

Both terms are non-positive 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 (92); 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

\[
(96) \quad z < \max_{\phi-\pi/2 < x < 0} \frac{\cos^{m-1}(x) \sin^{d-m-1}(x - \phi)}{\cos^{d-m-1}(x - \phi) \sin^{d-m-1}(x)}.
\]

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

\[
(97) \quad 1 < \frac{\cos(x)}{\cos(x - \phi)} \quad \text{and} \quad \frac{\sin(x - \phi)}{\sin(x)} < -1,
\]

so (96) implies \( z < -1 \), which violates our assumption. \[\square\]
5.3. Analysis of the Hessian Matrix at the Non-symmetric 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 non-global 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 \( \alpha_i \) are the same, \( n(\vec{\alpha}) \neq 0 \), and \( n_i(\vec{\alpha}) = 0 \) for all \( i \), then, up to permutation of the directions, the Hessian at \( \vec{\alpha} \) can be written as

\[
H = \frac{2(n(\vec{\alpha}))^2}{1 + \lambda} \begin{bmatrix}
(1 - \Theta^{-1})I_m + \Theta^{-1}I_m^{*} & I_m^{*}I_{d-m}^{*} \\
I_{d-m}I_{m}^{*} & (1 - \Theta)I_{d-m} + \Theta I_{d-m}I_{d-m}^{*}
\end{bmatrix},
\]

(98)

where \( \Theta = \cot(\alpha_1)\cot(\alpha_1 - \phi) \).

If \( m = 1 \) then \( 1 \neq \Theta \) and \( \Theta^{-1} \) cancels out. If \( m > 1 \) then \( 1 \neq \Theta \neq 0 \neq \Theta^{-1} \).

Proof. The Hessian depends on \( n(\vec{\alpha}) \) and \( n_{jk}(\vec{\alpha}) \), so we need to compute them. The procedure is to
1. start with the original formula (34) for \( n(\vec{\alpha}) \) and (36) or (37) for \( n_{jk}(\vec{\alpha}) \),
2. apply the structural constraint (77),
3. apply (78) in the form (88) and (89) to eliminate \( \alpha_d \),
4. apply (85) in the form

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

(100)

to eliminate \( z \), and
5. use trigonometric identities to simplify.

For \( n(\vec{\alpha}) \) we obtain

\[
\|T\|n(\vec{\alpha}) = \cos^m(\alpha_1) \cos^{d-m}(\alpha_d) + z \cos^m(\alpha_1 - \phi) \cos^{d-m}(\alpha_d - \phi)
\]

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

\[
= (-1)^{d-m} \sin^{d-m-1}(\alpha_1 - \phi) \cos^{m-1}(\alpha_1) (\cos(\alpha_1) \sin(\alpha_1 - \phi) - \cos(\alpha_1 - \phi) \sin(\alpha_1))
\]

\[
= (-1)^{d-m-1} \sin(\phi) \sin^{d-m-1}(\alpha_1 - \phi) \cos^{m-1}(\alpha_1).
\]

(101)

For \( n_{jk}(\vec{\alpha}) \) we have to split into cases based on the relationship of \( j, k, \) and \( m \). When \( j = k \), by (37) we have \( \|T\|n_{jk}(\vec{\alpha}) = -\|T\|n(\vec{\alpha}) = -(101) \). For \( j \leq m, k \leq m, \) and \( j \neq k \) we obtain

\[
\|T\|n_{jk}(\vec{\alpha}) = (-1)^{d-m} \sin(\phi) \frac{\sin(\alpha_1)}{\cos(\alpha_1 - \phi)} \sin^{d-m}(\alpha_1 - \phi) \cos^{m-2}(\alpha_1).
\]

(102)

For \( j > m, k > m, \) and \( j \neq k \) we obtain

\[
\|T\|n_{jk}(\vec{\alpha}) = (-1)^{d-m} \sin(\phi) \frac{\cos(\alpha_1 - \phi)}{\sin(\alpha_1)} \sin^{d-m-2}(\alpha_1 - \phi) \cos^{m}(\alpha_1).
\]

(103)
For \( j \leq m \) and \( k > m \), or \( j > m \) and \( k \leq m \), we obtain

\[
\|T\|n_{jk}(\vec{\alpha}) = (-1)^{d-m} \sin(\phi) \sin^{d-m-1}(\alpha_1 - \phi) \cos^{m-1}(\alpha_1) = -(101) = -\|T\|n(\vec{\alpha}).
\]

Starting from (45) and applying the assumption that \( n_i(\vec{\alpha}) = 0 \) for all \( i \), the entries in the Hessian \( H \) are given by

\[
H_{jk} = \frac{-2(n_k(\vec{\alpha})n_j(\vec{\alpha}) + n(\vec{\alpha})n_{jk}(\vec{\alpha}))}{1 + \lambda} = \frac{-2n(\vec{\alpha})n_{jk}(\vec{\alpha})}{1 + \lambda} = \frac{2(n(\vec{\alpha}))^2 - n_{jk}(\vec{\alpha})}{n(\vec{\alpha})}.
\]

To prove the lemma, it suffices to show that \(-n_{jk}(\vec{\alpha})/n(\vec{\alpha})\) has the block structure described for (98). If \( j = k \), by (37) we have \(-n_{jk}(\vec{\alpha})/n(\vec{\alpha}) = 1\) as desired. For \( j \leq m \), \( k \leq m \), and \( j \neq k \) we obtain

\[
\frac{-n_{jk}(\vec{\alpha})}{n(\vec{\alpha})} = \frac{-n(\vec{\alpha})}{n(\vec{\alpha})} = \Theta^{-1},
\]
as desired. For \( j > m \), \( k > m \), and \( j \neq k \) we obtain

\[
\frac{-n_{jk}(\vec{\alpha})}{n(\vec{\alpha})} = \frac{-n(\vec{\alpha})}{n(\vec{\alpha})} = \Theta,
\]
as desired. For \( j \leq m \) and \( k > m \), or \( j > m \) and \( k \leq m \), by (104) we have \(-n_{jk}(\vec{\alpha})/n(\vec{\alpha}) = 1\) as desired.

By Theorem 11, if \( m > 1 \) then \( \alpha_1 \in \{\phi - \pi/2, 0, \phi/\pi/2\} \) is never a solution, so \( \Theta \neq 0 \) and \( \Theta^{-1} \neq 0 \). If \( \Theta = 1 \), then \( \Theta = (99) = 1 \) yields \( \cos(2\alpha_1 - \phi) = 0 \), which implies \( \alpha_1 = \phi/2 \pm \pi/4 \). Then by (78) we have \( \alpha_d = \phi/2 \pm \pi/4 = \alpha_1 \). Since this contradicts (77), we know \( \Theta \neq 1 \).

**Lemma 13.** The Hessian (98) has eigenvalues \( 2(n(\vec{\alpha}))^2(1 + \lambda)^{-1} \) times

\[
\begin{align*}
1 - \Theta^{-1} & \quad \text{with multiplicity } m - 1, \\
1 - \Theta & \quad \text{with multiplicity } d - m - 1, \quad \text{and} \\
The eigenvalues of & \quad \begin{bmatrix} 1 + (m-1)\Theta^{-1} & \sqrt{m(d-m)} \\
\sqrt{m(d-m)} & 1 + (d-m-1)\Theta \end{bmatrix}.
\end{align*}
\]

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

\[
\begin{bmatrix} U_{mm} & 0 \\
0 & I_{d-m} \end{bmatrix} \begin{bmatrix} I_m & 0 \\
0 & U_{(d-m)1} \end{bmatrix} \frac{1 + \lambda}{2(n(\vec{\alpha}))^2} H \begin{bmatrix} I_m & 0 \\
0 & U_{(d-m)1} \end{bmatrix} \begin{bmatrix} U_{mm} & 0 \\
0 & I_{d-m} \end{bmatrix} = \begin{bmatrix} (1 - \Theta^{-1})I_m + \Theta^{-1}m e_m e_m^* & \sqrt{m\sqrt{d-m}} e_m^* \\
\sqrt{m\sqrt{d-m}} e_m & (1 - \Theta)I_{d-m} + \Theta(d-m) e_1 e_1^* \end{bmatrix}
\]

\[
= \begin{bmatrix} (1 - \Theta^{-1})I_{m-1} & 0 & 0 & 0 \\
0 & 1 + \Theta^{-1}(m-1) & \sqrt{m\sqrt{d-m}} & 0 \\
0 & \sqrt{m\sqrt{d-m}} & 1 + \Theta(d-m-1) & 0 \\
0 & 0 & 0 & (1 - \Theta)I_{d-m-1} \end{bmatrix}.
\]

Conjugation with unitary matrices preserves eigenvalues and by reading the eigenvalues off (111) 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 \( k \), \( n(\alpha) \neq 0 \), and \( n_i(\alpha) = 0 \) for all \( i \), then \( \alpha \) is a saddle point.

Proof. Let \( x_m \geq x_{m+1} \) denote the eigenvalues of (110). By computing the determinant and trace of (110), we have

\[
\begin{align*}
x_m x_{m+1} &= (2 - d) + (d - m - 1)\Theta + (m - 1)\Theta^{-1} \quad \text{and} \\
x_m + x_{m+1} &= 2 + (m - 1)\Theta^{-1} + (d - m - 1)\Theta.
\end{align*}
\]  

If \( \Theta < 0 \), then each summand in (112) 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 (112) 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 (112) and (113), 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 \( \mu/\eta \), where \( \mu \) is the smallest (most negative) and \( \eta \) 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 (14).

In subsection 5.4.1 we consider the case \( \Theta = 99 \). We can solve (85) for \( \alpha_1 \) and from there determine complete information about the saddles. We show that for \( z < 0 \) saddles with \( \mu/\eta \to 0^- \) exist as \( \phi \) approaches a certain nonzero angle, whereas for \( 0 < z \) the ratio \( \mu/\eta \) is bounded away from zero.

In subsection 5.4.2 we consider \( m > 1 \). We cannot solve (85) analytically, so we solve it numerically and then numerically determine \( \mu/\eta \). We observe that for \( z < 0 \) as \( \phi \) 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 \( \mu/\eta \) as a function of \( d \) and \( \Theta = (99) = \cot(\alpha_1) \cot(\alpha_1 - \phi) \) 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 \( \alpha_1 \) that yield a saddle, as a function of \( d, \phi, \) and \( z \). Thus for any configuration \( (d, \phi, z) \) we can determine \( \mu/\eta \). In Lemma 17 we show that \( \Theta = 1 \) corresponds to a symmetric, non-hyperbolic 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

\[
\mu = \frac{2(2 - d)}{(2 + (d - 2)\Theta) + \sqrt{(2 + (d - 2)\Theta)^2 - 4(d - 2)(\Theta - 1)}},
\]

which is increasing and has value \((2 - d)/2\) at \( \Theta = -1 \);
Table 3
Analysis of cases for Lemma 15

<table>
<thead>
<tr>
<th>( \Theta ) ∈</th>
<th>((-\infty, -1])</th>
<th>((-1, 1])</th>
<th>((1, \infty])</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_d = 1 - \Theta )</td>
<td>( x_d &gt; 0 )</td>
<td>( x_d &lt; 0 )</td>
<td>( x_d &lt; 0 )</td>
</tr>
<tr>
<td>by (118)</td>
<td>( x_1 x_2 &lt; 0 ) so ( x_1 &gt; 0 &gt; x_2 )</td>
<td>( x_1 x_2 &gt; 0 )</td>
<td>( x_1 x_2 &gt; 0 )</td>
</tr>
<tr>
<td>by (119)</td>
<td>( (x_1 - x_d)(x_2 - x_d) \geq 0 ) so ( x_d \geq x_1 &gt; x_2 )</td>
<td>( (x_1 - x_d)(x_2 - x_d) &lt; 0 ) so ( x_1 &gt; x_d &gt; x_2 )</td>
<td>( x_1 + x_2 &gt; 0 ) so ( x_1 \geq x_2 &gt; 0 )</td>
</tr>
<tr>
<td>Combined</td>
<td>( x_d \geq x_1 &gt; x_2 \geq x_2 )</td>
<td>( x_1 &gt; x_d &gt; x_2 \geq x_2 )</td>
<td>( x_1 \geq x_2 &gt; 0 &gt; x_d )</td>
</tr>
<tr>
<td>( \mu/\eta )</td>
<td>( x_2/ x_d )</td>
<td>( x_2/x_1 )</td>
<td>( x_d/x_1 )</td>
</tr>
</tbody>
</table>

for \(-1 \leq \Theta < 1\) the Hessian has \(d - 1\) positive eigenvalues and 1 negative eigenvalue, and

\[
\frac{\mu}{\eta} = \frac{(2 + (d - 2)\Theta) - \sqrt{(2 + (d - 2)\Theta)^2 - 4(d - 2)(\Theta - 1)}}{(2 + (d - 2)\Theta) + \sqrt{(2 + (d - 2)\Theta)^2 - 4(d - 2)(\Theta - 1)}},
\]

which is increasing and has value \((2 - d)/2\) at \(\Theta = -1\) and limit 0 as \(\Theta \to 1^-\); and

for \(1 < \Theta < \infty\) the Hessian has 2 positive eigenvalues and \(d - 2\) negative eigenvalues, and

\[
\frac{\mu}{\eta} = \frac{(2 + (d - 2)\Theta) - \sqrt{(2 + (d - 2)\Theta)^2 - 4(d - 2)(\Theta - 1)}}{2(2 - d)},
\]

which is decreasing and has limit 0 as \(\Theta \to 1^+\).

**Proof.** When \(m = 1\), we no longer have the eigenvalue \(1 - \Theta^{-1}\) but still have the eigenvalue \(x_d = 1 - \Theta\) with multiplicity \(d - 2\). By (110), \(x_1\) and \(x_2\) are eigenvalues of

\[
\begin{bmatrix}
1 & \sqrt{d - 1} \\
\sqrt{d - 1} & 1 + (d - 2)\Theta
\end{bmatrix},
\]

from which we can compute

\[
\text{det (117)} = x_1 x_2 = (d - 2)(\Theta - 1),
\]

\[
\text{det (117) - (1 - \Theta)I} = (x_1 - x_d)(x_2 - x_d) = (d - 1)(\Theta^2 - 1), \quad \text{and}
\]

\[
\text{trace (117)} = x_1 + x_2 = 2 + (d - 2)\Theta.
\]

To determine \(\mu/\eta\), we need to determine the signs and order of the eigenvalues. We organize the cases in Table 3. We can directly compute the eigenvalues

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

and the required ratios \(x_2/x_d\), \(x_2/x_1\), and \(x_d/x_1\). By (118) 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 (114)–(116). The values at \(\Theta = -1\) are determined by plugging into (114) and (115). The
limits as \( \Theta \to 1^\pm \) are obtained by plugging into (115) and (116). To determine the intervals of increase and decrease, we compute the derivative of \( \mu/\eta \). Since the computation is routine but long, we omit it.

**Lemma 16.** For \( m = 1 \), the saddles are located at

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

where \( S = 1 \) if \( d \) is odd, two solutions with \( S = \pm 1 \) exist if \( d \) is even and \( z < 0 \), and no solutions exist (by Theorem 11) if \( d \) is even and \( z > 0 \).

**Proof.** For \( m = 1 \), (85) becomes

\[
0 = \sin(d-2)(\alpha_1 - \phi) + z \sin(d-2)(\alpha_1),
\]

which we can solve for \( z \) to obtain

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

If \( z > 0 \) and \( d \) is even then there is no solution. Otherwise we note that \( \cot(\alpha_1) \) is one-to-one and has range \((-\infty, \infty)\), so there is exactly one solution when \( d \) is odd and two solutions corresponding to signs \( \pm 1 \) when \( d \) is even. To treat these cases together, let \( S = 1 \) for \( d \) odd and \( S = \pm 1 \) for \( d \) even. Solving (124) for \( \alpha_1 \) yields (122).

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

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

and \((d, z, \phi)\) yields a non-hyperbolic, symmetric stationary point located at \( \vec{\alpha} = (\phi/2 + \pi/4) \vec{1} \) with \( d - 1 \) zero eigenvalues and a single positive eigenvalue. Moreover, (125) 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 \( \vec{\alpha} = \alpha \vec{1} \) 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 \((x_2 \text{ and } x_d)\) 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 (124) yields

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

By our assumption \(|z| \leq 1\) we must choose \( \pm \mapsto + \) and by our assumption \( \phi \in (0, \pi/2) \) we must have \(-1 < z < 0\). Since (125) \( \mapsto -1^+ \) as \( \phi \mapsto 0^+ \) and (125) \( \mapsto 0^- \) as \( \phi \mapsto (\pi/2)^- \), we see (125) 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 (124) for \( \cot(\alpha_1) \) yields \( \cot(\alpha_1) = (\cos(\phi) + z^{1/(d-2)})/\sin(\phi) \). Similarly, in (124), 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

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

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

In Figure 8 we show \( \alpha_1, \alpha_d, \) and \( \mu/\eta \) 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 \( \mu/\eta < -1 \). We also plotted, but do not include, the \( d = 6 \) case, which has no solutions when \( 0 < z \) and two solutions when \( z < 0 \). We find that the values corresponding to the \( S = 1 \) solution are similar to the \( d = 5 \) case with the same \( z \) and the values correspond to the \( S = -1 \) solution are similar to the \( d = 5 \) case with \( |z| \). For \( z = -1/2 \), we can see where the \( \alpha_1 \) and \( \alpha_d \) curves cross, yielding the non-hyperbolic, symmetric stationary point in Lemma 17. This point shows up in Figure 5 as a discontinuity in \( \hat{h}(G_1) \).

5.4.2. Saddles with \( m > 1 \). For \( m > 1 \) we are unable to solve (85) analytically for \( \alpha_1 \), so we cannot conduct a full analysis. We did numerically find solutions to (85) as indicated in Table 2, and from them compute \( \mu/\eta \), but we cannot be certain that we captured all phenomenon. From testing various combination of \( (d, m, z) \), we observe the following:

- For \( z > 0 \), there are cases with \( -1 < \Theta \), in contrast to Lemma 18. However \( \Theta \) is never close to 0.
- For \( z < 0 \), \( m > 1 \), and \( d \) odd, when \( \phi \) is sufficiently small there are 3 solutions, 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 \( \phi \) is sufficiently small there are 4 solutions rather than 2.
- We can observe directly from Lemma 13 that only \( \Theta = 1 \) can yield a non-hyperbolic stationary point. Lemma 17 shows this symmetric stationary point exists as a continuous extension of a \( m = 1 \) non-symmetric saddle. We find that if \( z < 0 \) then this symmetric stationary point also exists as a continuous extension of non-symmetric saddles for every \( 1 < m \leq d/2 \). Thus for parameter configurations near that identified in Lemma 17 there are many bad saddles, especially taking into account the possible permutations of directions. We illustrate this situation in Figure 9.

6. Analysis with Partially Symmetric \( T \) and \( G_1 \). In sections 4 and 5 we only considered symmetric \( T \), which have \( \phi = \phi^\dagger \). Analysis with general \( \phi \) 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 non-hyperbolic stationary points still occur and so is not specific to the symmetric case.
Figure 8. 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 $\mu/\eta$ with vertical range $[-1, 0]$; when $\mu/\eta < -1$ the algorithm would escape in one or two steps.
By inserting into the gradient (44), 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 (45), simplifying as in Lemma 12, and transforming as in Lemma 13. The
Figure 10. Locations of the non-hyperbolic 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 $\phi_0$ marked with a dashed line and $\pi/2$ with a solid line. Similarly, the vertical axis is $\phi_d$. When either angle exceeds $\pi/2$ we can reflect it by $\phi \mapsto \pi - \phi$ and set $z = -1$.

The eigenvalues are $2(n(\vec{a}))^2(1 + \lambda)^{-1}$ times

\begin{align}
(128) & \quad 1 + \tan^2(\phi_1/2) \text{ with multiplicity } m - 1, \\
(129) & \quad 1 + \tan^2(\phi_d/2) \text{ with multiplicity } d - m - 1, \text{ and the eigenvalues of} \\
(130) & \quad \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}.
\end{align}

Since (128) and (129) are positive, this stationary point is stable transverse to the partial symmetry. The bifurcation value corresponding to $\phi_0$ in (66) occurs when the determinant of (130) is zero, which is when

\begin{align}
(131) & \quad \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).
\end{align}

In Figure 10 we illustrate this relationship between $\phi_1$ and $\phi_d$.

REFERENCES


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


With comments by Jan de Leeuw, Pieter M. Kroonenberg and Donald S. Burdick and a rejoinder by the authors.


E. Pruliere, F. Chienesta, and A. Ammar, On the deterministic solution of multidimensional parametric models using the Proper Generalized Decomposition, Mathematics and Computers in Simu-
Fitting a rank-2 tensor with a rank-1 tensor


[110] A. Stegeman and T. T. T. Lam, Improved uniqueness conditions for canonical tensor decompositions


