“Nahm-like” gradient flows: analysis and implementation in Matlab

Andre Mas
James Madison University

June 29, 2022

Abstract

The central interest in [3] is a Lie algebra gradient flow that describes the geometry of a nilpotent variety. These gradient flows are seen to be a generalization of Nahm's equations, which were historically used in articles such as [1] and [2] to construct and classify monopoles.

We analyze a parameterized family of these "Nahm-like flows", giving a characterization of critical points of the flows (zeros of their respective vector fields). This is done in the case of $\mathfrak{g} = \mathfrak{so}(3)$, which is isomorphic to $\mathbb{R}^3$ with the Lie bracket being the standard cross product. We show that "diagonal" trajectories necessarily stay diagonal along these flows, which allows for a dimensional reduction from $\mathbb{R}^9$ to $\mathbb{R}^3$. For the flow analyzed in [3], we present an ansätz diagonal solution that converges to a nontrivial zero of the flow's associated vector field.

This project made extensive use of the visualization tools in Matlab to motivate analytic results. This follows the trend of "experimental mathematics" that has gained prominence in recent years, in which computer software and numerical methods are used to guess and later verify solutions to problems in pure mathematics.

A Matlab script to visualize the aforementioned ansätz solution is provided. Implementation of standard gradient descent techniques is also included, with the ability to modify parameters and initial conditions. Additionally, we propose a modified gradient descent algorithm to detect trajectories that approach nontrivial zeros. Documentation is included in each of these files, which are included in the typical Matlab .m format.

Acknowledgements

This project was conducted over the Summer of 2022 alongside Dr. David Duncan at James Madison University. I am highly grateful for his patience and guidance during this process. Thank you to the numerous other professors and faculty members who have contributed to my undergraduate education, and introduced me to the methods and ideas used during this project.

Contents

1 Introduction and analytic results **3**
1.1 "Nahm-like" gradient flows 3
1.2 Diagonal trajectories 4
1.3 Zeros of the flow 5
1.4 Solutions of the flow 7

2 Implementation in Matlab **9**
2.1 Revisited: Visualizing the exact solution to Kronheimer's flow . . 9
2.2 Gradient descent 10
2.3 Velocity based gradient descent 10

Chapter 1

Introduction and analytic results

1.1 "Nahm-like" gradient flows

Let $\mathfrak{g}$ be a Lie algebra equipped with an Ad-invariant inner product $\langle\ ,\ \rangle$. The initial function in [3] is $\varphi : \mathfrak{g} \times \mathfrak{g} \times \mathfrak{g} \to \mathbb{R}$ given by:

$$\varphi(A_1, A_2, A_3) \stackrel{\text{def}}{=} \sum_{i=1}^{3} \langle A_i, A_i \rangle + \big\langle A_1, [A_2, A_3] \big\rangle$$

The function φ is a relatively simple construction, given by the sum of two functions f and ψ, where:

$$f(A_1, A_2, A_3) \stackrel{\text{def}}{=} \sum_{i=1}^{3} \langle A_i, A_i \rangle$$

$$\psi(A_1, A_2, A_3) \stackrel{\text{def}}{=} \big\langle A_1, [A_2, A_3] \big\rangle$$

The generalization of φ that we will consider is constructed by adding a family of three real valued parameters $\alpha = \{\alpha_1, \alpha_2, \alpha_3\}$ to f, and a single parameter β to ψ, giving the functions:

$$f_\alpha(A_1, A_2, A_3) \stackrel{\text{def}}{=} \sum_{i=1}^{3} \alpha_i \langle A_i, A_i \rangle$$

$$\psi_\beta(A_1, A_2, A_3) \stackrel{\text{def}}{=} \beta \big\langle A_1, [A_2, A_3] \big\rangle$$

We then set $\hat{\varphi}$ to be the sum of f_α and ψ_β. Let $A = (A_1, A_2, A_3)$, so that $\hat{\varphi}$ can be interpreted as a function defined on $\mathfrak{g}^3$. The gradient flow equation $\dot{A} = -\nabla \hat{\varphi}(A)$ becomes a nonlinear system of three equations given by:

$$\dot{A}_i = -2\alpha_i A_i - \beta[A_{i+1}, A_{i+2}] \tag{1.1}$$

This vector field has zero curl, and zeros necessarily have orthogonal components. The divergence of the vector field is equal to the trace of its linear component:

$$\mathrm{div}\Big(\nabla\hat{\varphi}\Big) = -2\sum_{i=1}^{3}\alpha_i$$

Remark 1.1.1. Note that the gradient flow analyzed in [3] is a special case of the flow in equation 1.1 that arises when each of the parameters $\alpha_i = \beta = 1$. In the case of $\alpha_i = 0$ and $\beta = 1$, equation 1.1 becomes Nahm's system [5] taking values in $\mathfrak{g}$.

The initial value problem for $\dot{A} = -\nabla\hat{\varphi}(A)$ that we shall consider is given by an initial condition at $t = 0$:

$$\begin{cases} \dot{A} = -\nabla\hat{\varphi}(A) \\ A_0 = A(0) \end{cases} \tag{1.2}$$

We now turn to the issue of existence and uniqueness for this initial value problem. The flow is Lipschitz, and we assume that the vector field $-\nabla\hat{\varphi}$ is continuous in some region $\mathbf{R}$ of A_0 defined by $\|A - A_0\| \leq \epsilon_1$, where ϵ_1 is greater than zero and $\|\cdot\|$ is the standard L^2 norm on $\mathfrak{g}^3$. Let $\epsilon_2 = \max\|\nabla\hat{\varphi}(A)\|$ on $\mathbf{R}$. It can be shown that ϵ_2 is bounded entirely in terms of ϵ_1, where a (not necessarily sharp) bound is given by:

$$\epsilon_2 \leq 4\max_{\mathrm{i}}|\alpha_i|\,\big(\epsilon_1 + \|A_0\|\big) + 4\sqrt{3}\,|\beta|\,\big(\epsilon_1 + \|A_0\|\big)^2$$

From the above, the existence of a unique solution is guaranteed on the interval $[0, \epsilon_1/\epsilon_2)$ by the Picard-Lindelöf theorem as stated in [4].

1.2 Diagonal trajectories

For the remainder of this project we specialize to the case of $\mathfrak{g} = \mathbb{R}^3$, with the Lie bracket being the standard cross product. To simplify various computations, we use a nonstandard basis for $\mathbb{R}^3$ that satisfies $-2v_i = v_{i+1}\times v_{i+2}$. For definiteness, we take:

$$v_1 = \begin{bmatrix} -2 \\ 0 \\ 0 \end{bmatrix} \quad v_2 = \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix} \quad v_3 = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$$

This flow takes place in $\mathbb{R}^9$. Through a slight abuse of notation, we imagine the vector field $-\nabla\hat{\varphi}$ acting on the 3×3 matrix $A = [A_1|A_2|A_3]$. We can then consider diagonal trajectories of the flow, and have the following lemma:

Lemma 1.2.1. *Suppose A is a diagonal matrix given by the above construction. Then $-\nabla\hat{\varphi}(A)$ is also diagonal.*

This asserts that diagonal trajectories remain diagonal for all time along the flow. The proof of lemma 1.2.1 is a straightforward computation:

Proof. Let $A = [A_1|A_2|A_3]$ be diagonal, where:

$$A_1 = \begin{bmatrix} x \\ 0 \\ 0 \end{bmatrix} \quad A_2 = \begin{bmatrix} 0 \\ y \\ 0 \end{bmatrix} \quad A_3 = \begin{bmatrix} 0 \\ 0 \\ z \end{bmatrix}$$

It may be verified that $-\nabla\hat{\varphi}(A)$ is diagonal, with the form:

$$\dot{A} = \begin{bmatrix} -2\alpha_1 x - \beta yz & 0 & 0 \\ 0 & -2\alpha_2 y - \beta zx & 0 \\ 0 & 0 & -2\alpha_3 z - \beta xy \end{bmatrix}$$

□

Remark 1.2.1. When working with diagonal trajectories, it will be convenient to think of these as three-tuples (x, y, z) in $\mathbb{R}^3$ rather than as diagonal matrices $\mathrm{diag}(x, y, z)$ in $\mathbb{R}^9$. This slight abuse of notation reduces the dimension of the flow considerably.

1.3 Zeros of the flow

We now discuss the zeros of the flow. That is, the triples (A_1, A_2, A_3) which satisfy:

$$-2\alpha_i A_i = \beta[A_{i+1}, A_{i+2}]$$

The trivial zero $(0, 0, 0)$ can be verified to be asymptotically stable. This follows from f_α having negative real eigenvalues, and the extent of the non-linearity from ψ_β being sufficiently well behaved. It will be useful in the upcoming analysis to have a characterization of nontrivial zeros. Continuing in the case of $\mathbb{R}^3$, we have the following theorem:

Theorem 1.3.1. *Suppose $\mathfrak{g} = \mathbb{R}^3$, $\alpha_i > 0$, and $\beta \neq 0$. Then the zero locus of the flow is given by:*

$$\left\{ g\Big(\frac{\sqrt{\alpha_2\alpha_3}}{\beta}v_1, \frac{\sqrt{\alpha_3\alpha_1}}{\beta}v_2, \frac{\sqrt{\alpha_1\alpha_2}}{\beta}v_3\Big) \;\Big|\; g \in SO(3) \right\} \cup \left\{ (0, 0, 0) \right\}$$

Proof. Since $-2\alpha_i A_i = \beta[A_{i+1}, A_{i+2}]$, the set $A = \{A_1, A_2, A_3\}$ forms an orthogonal basis for $\mathbb{R}^3$. It follows that there exists some $SO(3)$ element g mapping $A_i \mapsto c_i v_i$, where the c_i are fixed scalars. This gives the system of equations:

$$\begin{aligned}
-2\alpha_1 c_1 v_1 &= \beta(c_2 v_2 \times c_3 v_3) \\
-2\alpha_2 c_2 v_2 &= \beta(c_3 v_3 \times c_1 v_1) \\
-2\alpha_3 c_3 v_3 &= \beta(c_1 v_1 \times c_2 v_2)
\end{aligned}$$

Since $-2v_i = v_{i+1} \times v_{i+2}$, this system reduces to solving for the coefficients c_i in terms of the parameters α_1, α_2, α_3, and β. We have:

$$\begin{aligned}
\alpha_1 c_1 &= \beta c_2 c_3 \\
\alpha_2 c_2 &= \beta c_3 c_1 \\
\alpha_3 c_3 &= \beta c_1 c_2
\end{aligned}$$

Algebraic manipulation gives that if each $\alpha_i > 0$ and $\beta \neq 0$, then:

$$c_i = \frac{\sqrt{\alpha_{i+1}\alpha_{i+2}}}{\beta}$$

Along with the zero solution, this gives the zero locus:

$$\left\{ g\left(\frac{\sqrt{\alpha_2\alpha_3}}{\beta} v_1, \frac{\sqrt{\alpha_3\alpha_1}}{\beta} v_2, \frac{\sqrt{\alpha_1\alpha_2}}{\beta} v_3 \right) \;\middle|\; g \in SO(3) \right\} \cup \left\{ (0,0,0) \right\}$$

□

Considering only diagonal trajectories, we have an exact characterization of the zero locus. Since the only diagonal elements of $SO(3)$ are given by:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

It follows as a corollary of theorem 1.3.1 that the only non-trivial zeros that are diagonal are the above $SO(3)$ elements multiplied by the matrix:

$$\left[\frac{\sqrt{\alpha_2\alpha_3}}{\beta} v_1 \;\middle|\; \frac{\sqrt{\alpha_3\alpha_1}}{\beta} v_2 \;\middle|\; \frac{\sqrt{\alpha_1\alpha_2}}{\beta} v_3 \right]$$

Corollary 1.3.1. *Let $\mathfrak{g} = \mathbb{R}^3$, with associated Lie group $SO(3)$. Let $\alpha_i > 0$ and $\beta \neq 0$. Then the diagonal zero locus consists of the following five matrices:*

$$
\begin{aligned}
&\mathrm{diag}\Big(-2\frac{\sqrt{\alpha_2\alpha_3}}{\beta}, 2\frac{\sqrt{\alpha_3\alpha_1}}{\beta}, 2\frac{\sqrt{\alpha_1\alpha_2}}{\beta}\Big)\\
&\mathrm{diag}\Big(2\frac{\sqrt{\alpha_2\alpha_3}}{\beta}, -2\frac{\sqrt{\alpha_3\alpha_1}}{\beta}, 2\frac{\sqrt{\alpha_1\alpha_2}}{\beta}\Big)\\
&\mathrm{diag}\Big(2\frac{\sqrt{\alpha_2\alpha_3}}{\beta}, 2\frac{\sqrt{\alpha_3\alpha_1}}{\beta}, -2\frac{\sqrt{\alpha_1\alpha_2}}{\beta}\Big)\\
&\mathrm{diag}\Big(-2\frac{\sqrt{\alpha_2\alpha_3}}{\beta}, -2\frac{\sqrt{\alpha_3\alpha_1}}{\beta}, -2\frac{\sqrt{\alpha_1\alpha_2}}{\beta}\Big)\\
&\mathrm{diag}\Big(0, 0, 0\Big)
\end{aligned}
$$

Example 1.3.1. Setting $\alpha_i = \beta = 1$, we have Kronheimer's flow from [3]. This gives the following zero locus:

$$
\begin{bmatrix} -2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \quad
\begin{bmatrix} 2 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \quad
\begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & -2 \end{bmatrix}, \quad
\begin{bmatrix} -2 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -2 \end{bmatrix}, \quad
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}
$$

1.4 Solutions of the flow

Let $\alpha_i > 0$ and $\beta \neq 0$. The following three diagonal trajectories can be verified to solve the flow and converge to $(0, 0, 0)$ as $t \to \infty$:

$$
(x_0 e^{-2\alpha_1 t}, 0, 0)\ , \quad (0, y_0 e^{-2\alpha_2 t}, 0)\ , \quad (0, 0, z_0 e^{-2\alpha_3 t})
$$

While there are no general methods for finding trajectories that converge to nontrivial zeros, assuming local symmetries of the vector field can lead to ansätz solutions.

Example 1.4.1. Let $\mathfrak{g} = \mathbb{R}^3$, and each of $\alpha_i = \beta = 1$. This is the flow analyzed in [3]. We restrict to diagonal trajectories. The goal is to find a trajectory that converges to a non-trivial zero as $t \to \infty$. We focus specifically on trajectories converging to $\mathrm{diag}(-2, 2, 2)$. Assuming a symmetry of the y and z components about this critical point, the flow reduces to:

$$
\begin{aligned}
\dot{x} &= -2x - y^2 \\
\dot{y} &= -2y - xy
\end{aligned}
$$

It may be shown that with the additional yz symmetry and as a result of the above equations, x and y are forced to satisfy the following relation:

$$
x^2 - y^2 = Ce^{-4t}
$$

Substituting y^2 in the equation for $\dot{x}$ gives a Ricatti ODE which may be solved directly. This leads to the following solution which converges to $(-2, 2, 2)$ as $t \to \infty$:

$$x = -2\mu\frac{e^{2\mu}+1}{e^{2\mu}-1} \quad y = \frac{4e^{\mu}\mu}{e^{2\mu}-1} \quad z = \frac{4e^{\mu}\mu}{e^{2\mu}-1}$$

where $\mu = e^{-2t}$.

Chapter 2

Implementation in Matlab

The primary focus of chapter 2 is on the numerical implementation of the theory developed in chapter 1. Using the result of lemma 1.2.1 that diagonal trajectories are preserved under the flow for $\mathfrak{g} = \mathbb{R}^3$, computer visualization in three spacial dimensions is possible. This was used to provide insight into the analytic behavior of these flows, particularly in the case of Kronheimer's flow with $\alpha_i = \beta = 1$.

Gradient flows are in many ways nicely behaved as ODE's. This is a result of the fact that a standard discretization scheme already exists in the form of gradient descent, meaning that typical ODE approximation schemes such as Runge-Kutta methods are not necessary to analyze these flows numerically. For this reason, the numerical implementation throughout this project was largely done with gradient descent.

This chapter is divided into three sections, each of which are paired with a Matlab .m file that provides implementation of the topic at hand. These files are included in a .zip formatted download of this project and labelled with the corresponding section number.

2.1 Revisited: Visualizing the exact solution to Kronheimer's flow

We now provide a visualization of the exact solution to Kronheimer's flow with $\alpha_i = \beta = 1$ presented in example 1.4.1. We first plot the diagonal zero locus from example 1.3.1 in cyan, and connect these points by lines as a visual aid. The trivial zero $(0, 0, 0)$ is plotted in pink. The exact solution trajectory is then plotted in red, and we may see that this trajectory indeed converges to $(-2, 2, 2)$.

2.2 Gradient descent

Various numerical methods are used in the literature to approximate solutions to gradient flows. Gradient descent is a standard discretization given by:

$$A_{n+1} = A_n - h\nabla\hat{\varphi}(A_n) \tag{2.1}$$

where n denotes the iteration number and h is a fixed step size.

Remark 2.2.1. Floating point error tends to accumulate as these iterations approach a zero, and inaccurate approximations may be obtained even for exceedingly small step sizes.

Implementation of the standard gradient descent algorithm is provided. We again use the example of Kronheimer's flow, with the zero locus plotted in cyan as a visual aid. Even with a starting point on the exact trajectory found in example 1.4.1, notice that round-off error may cause the approximation to either diverge or converge to $(0, 0, 0)$. This occurs even for exceedingly small step-sizes.

2.3 Velocity based gradient descent

To avoid the issue outlined in remark 2.2.1, we define a velocity metric V as follows:

$$V = \frac{\|A_{n+1} - A_n\|}{h} = \|\nabla\hat{\varphi}(A_n)\|$$

Notice that $V \to 0$ as the iterates A_n approach a zero of the flow. By setting a "velocity tolerance" V_{tol}, we can detect when the gradient descent iterations enter a sufficiently small neighborhood of a zero. This gives a stopping criterion for the iterations to avoid accumulation of floating point error. We recommend a starting velocity tolerance of $V_{tol} = 1/2$.

Implementation of this algorithm is included. In the unedited script's example, notice that starting along the exact trajectory found in example 1.4.1 no longer diverges or converges to $(0, 0, 0)$ as seen with standard gradient descent. This script also outputs the maximum time before the velocity tolerance is met.

Bibliography

[1] N. J. Hitchen, *On the construction of monopoles.* (1983).

[2] S. Donaldson, *Nahm's Equations and the Classification of Monopoles.* (1984).

[3] P. Kronheimer, *Instantons and the Geometry of the Nilpotent Variety.* (1990).

[4] J. Liu, *A First Course in the Qualitative Theory of Differential Equations.* (2003).

[5] B. Charbonneau et al, *Construction of Nahm data and BPS monopoles with continuous symmetries.* (2022).