

PROJECTIVE PATH TRACKING FOR HOMOTOPY CONTINUATION METHOD

By

Tianran Chen

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Applied Mathematics

2012

ABSTRACT

PROJECTIVE PATH TRACKING FOR HOMOTOPY CONTINUATION METHOD

By

Tianran Chen

Solving systems of polynomial equations is an important problem in mathematics with a wide range of applications in many fields. The homotopy continuation method is a large class of reliable and efficient numerical methods for solving systems of polynomial equations. An essential component in the homotopy continuation method is the path tracking algorithm for tracking smooth paths of one real dimension. “Divergent paths” pose a tough challenge to path tracking algorithms as the tracking of such paths are generally impossible. The existence of such paths is, in part, caused by $\mathbb{C}^n$, the space in which homotopy methods usually operate, being non-compact. A well known remedy is to operate inside the complex projective space instead. Path tracking inside the complex projective space is the focus of this article. An existing method based on the use of generic “affine charts” of the complex projective space is widely used. While it works well in theory, we will point out the unnecessary numerical instability it could potentially create via the analysis of “path condition”. This article, then proposes a numerically superior approach for projective path tracking developed from the point of view of the Riemannian geometry of the complex projective space.

Copyright by
TIANRAN CHEN
2012

This dissertation is dedicated in loving memory of my dear A'po.

ACKNOWLEDGMENTS

This dissertation would not have been possible without the help from many people around me, to only some of whom it is possible to give particular mention here.

Above all, I would like to thank my advisor Professor Tien-Yien Li for his guidance, advice, understanding, support, and patience, not to mention his unsurpassed knowledge in every aspect of the Polyhedral homotopy continuation method. However, more importantly, his deep appreciation to an unusually wide range of mathematical subjects and the basic instinct of constantly asking why will certainly benefit me for years to come.

I am very grateful to Professor Zhonggang Zeng and Professor Tsung-Lin Lee for the discussions on various topics in numerical analysis which has greatly influenced my views on many technical aspects of the homotopy continuation methods. I would also like to thank my fellow graduate students Chaitanya Senapathi and Faramarz Vafaei for their helps during the project. Nick Ovenhouse, a research assistant in our group, has provided great helps in many aspects of the implementation of the projective path tracking algorithms as well as the creation of the tools for creating graphs included in this article of which I am very grateful. I would like to acknowledge the financial and technical support from the Michigan State University.

Last, but by no means least, I thank my family for their love and support for which my mere expression of thanks does not suffice.

TABLE OF CONTENTS

List of Tables	viii
List of Figures	ix
Chapter 1 Introduction	1
1.1 The basics of homotopy continuation methods for solving polynomial systems	1
1.2 Basic affine path tracking algorithms	3
1.3 Numerical condition of paths	4
1.3.1 Why path condition matters	6
1.3.2 Ill-conditioned paths	7
1.3.2.1 Divergent paths	7
1.3.2.2 Paths with large coordinates	8
1.3.2.3 Paths approaching singular points in the middle	8
1.3.2.4 Paths approaching singular end points	9
1.3.2.5 Summary of ill-conditioned paths	9
Chapter 2 Projective Path Tracking	10
2.1 Background: working in projective space	11
2.1.1 Projective path tracking using affine charts	14
2.1.2 Numerical analysis of the homotopy path	18
2.2 Geometry of the complex projective space	22
2.2.1 Identification between $\mathbb{C}P^n$ and $\mathbb{R}^{2n+1}$	22
2.2.2 Notation and terminology	25
2.2.3 Realizing $\mathbb{C}P^n$ as S^{2n+1}/U	25
2.2.4 Tangent formula for $\mathbb{C}P^n$	29
2.2.5 Distance formula for $\mathbb{C}P^n$	30
2.2.5.1 Riemannian geodesics	31
2.2.5.2 Distance on S^1 and S^{2n+1}	35
2.2.5.3 Closest representatives of two points in $\mathbb{C}P^n$	36
2.2.5.4 Geodesics in $\mathbb{C}P^n$	39
2.2.5.5 Summary: the distance formula in $\mathbb{C}P^n$	42
2.3 Projective predictors	43
2.3.1 Projective Euler's method	43
2.4 Projective correctors	44
2.4.1 Projective Newton's method	45
2.4.1.1 Convergence property	46
2.4.1.2 Stopping criteria	46

2.4.2	Projective Newton's method with damping factor	47
2.5	Projective Newton's homotopy	48
2.6	Numerical analysis of the algorithms	50
2.7	Numerical determination of residual	52
2.7.1	Affine residual	53
2.7.2	Tangential residual	54
2.7.3	Spherical residual	55
2.7.4	Summary of residuals	55
2.8	Special conditioning	56
2.8.1	The curious case of barry	56
2.8.2	The general problem	58
2.8.3	Dynamic row-scaling	59
2.9	The path tracking algorithm	61
2.10	Numerical results	62
Chapter 3	Endgame	65
3.1	Local theory of homotopy paths	66
3.1.1	Background: local theory of holomorphic varieties	67
3.1.2	Local normal form of affine homotopy paths	71
3.1.3	Local normal form of projective homotopy paths	73
3.2	Projective endgame based on Cauchy integral	77
3.2.1	Stopping criteria based on Riemannian distance	80
3.2.2	Stopping criteria based on tangent vector	82
3.2.3	Consistency tests	83
Bibliography		85

LIST OF TABLES

Table 2.1	Average percentage, rounded to the second digit after the decimal point, of paths with large maximum path condition number over 10 different runs when the homotopy $\hat{H}^L = \mathbf{0}$ of Equation (2.2) is used to solve different polynomial systems (all divergent paths and those with singular end points are ignored). At first glance, the percentages seem rather small, however one must keep in mind that hundreds of thousands of paths are tracked in solving some of the systems, so even a very small percentage will translate to a large absolute number of paths with very high maximum path condition.	20
Table 3.1	The scales of $ x_0 $ at end points of some paths that are known to be “at infinity”: These paths comes from solving systems listed on the first column using the projective path tracking we have discussed. .	66
Table 3.2	Comparison of the magnitude of $ x_0 $, in $\log_{10}$ scale, with and without the Cauchy integral method. The second column shows the winding number of the loop used for evaluating Cauchy integral. The third column shows the approximate order of magnitude of $ x_0 $ at the end point using the projective path tracking alone, while the last column shows the same measure but with Cauchy integral producing the end point.	80

LIST OF FIGURES

Figure 2.1	The path condition number, in $\log_{10}$ scale, of a single path defined by Equation (2.2) $\hat{H}^L = \mathbf{0}$ for solving the <code>cyclic7</code> problem: the path condition can be as large as 10^{16} “in the middle” of the path tracking.	16
Figure 2.2	The path condition numbers, in $\log_{10}$ scale, of a path defined by Equation (2.2) $\hat{H}^L = \mathbf{0}$ (shown in solid lines) and that of the affine associate of the same projective path (shown in dashed line) defined by the original homotopy $H = \mathbf{0}$ tracked for solving the <code>cyclic7</code> problem. The two paths represent the same projective path in $\mathbb{CP}^n \times [0, 1]$ defined by $\hat{H} = \mathbf{0}$, but the difference in their respective path condition numbers is striking!	17
Figure 2.3	The histogram of maximum path condition in $\log_{10}$ scale for all paths defined by the original homotopy $H = \mathbf{0}$ (in striped bars) and those defined by the homotopy $\hat{H}^L = \mathbf{0}$ of Equation (2.2) (in solid bars) for solving the <code>cyclic7</code> problem. The height of each bar indicates the total number of paths having the corresponding maximum path condition. For example the leftmost striped bar has a height of 852 and occupies a horizontal interval near 2. This bar signifies that there are 852 paths defined by the original $H = \mathbf{0}$ having maximum path condition approximately in the scale of 10^2 . This graph represents the average results of 10 different runs. While the two approaches are essentially tracking the different projections of the same set of projective paths defined by $\hat{H} = \mathbf{0}$, the difference in maximum path condition distribution is indeed striking!	21
Figure 2.4	Path condition, in $\log_{10}$ scale, along a single path tracked for solving the <code>barry</code> problem using projective path tracking algorithms on S^{2n+1}/U . The maximum path condition exceeds 10^9 .	56
Figure 2.5	The 2-norm of rows, in $\log_{10}$ scale, of the Jacobian matrix along a single path defined by $\hat{H} = \mathbf{0}$, tracked using projective path tracking algorithms on S^{2n+1}/U for solving the <code>barry</code> problem.	57
Figure 2.6	The comparison between path condition of a single path tracked in solving the <code>barry</code> problem using projective path tracking with (dashed) and without (solid) the dynamic row-scaling	60

Figure 2.7	The flowchart summarizes the overall projective path tracking algorithm	62
Figure 2.8	The histogram shows the distribution of maximum path conditions along all paths tracked for solving the <code>cyclic7</code> problem using projective path tracking algorithms on S^{2n+1}/U together with the dynamic row-scaling technique.	63
Figure 2.9	The histogram shows the distribution of maximum path conditions along all paths tracked for solving the <code>cyclic7</code> problem with the two different methods. The height of each solid bar represents the number of paths having the corresponding maximum path condition when the traditional method with affine charts described in Equation (2.2) is used. The striped bars show the same information when the projective path tracking algorithms on S^{2n+1}/U developed in this chapter is used. As with previous cases, this graph represents the average result of 10 different runs.	64

Chapter 1

Introduction

1.1 The basics of homotopy continuation methods for solving polynomial systems

Solving systems of polynomial equations is an important problem in mathematics. It has a wide range of applications in many fields of mathematics, sciences, and engineering. In this article we restrict our attention to systems of n polynomial equations in n variables with complex coefficients of the form

$$P(\mathbf{x}) = \begin{cases} p_1(x_1, \dots, x_n) = \sum_{k_1, \dots, k_n} c_{k_1, \dots, k_n}^{(1)} x_1^{k_1} \cdots x_n^{k_n} = 0 \\ \vdots \\ p_n(x_1, \dots, x_n) = \sum_{k_1, \dots, k_n} c_{k_1, \dots, k_n}^{(n)} x_1^{k_1} \cdots x_n^{k_n} = 0 \end{cases}$$

which will simply be called **polynomial systems**. By the Abel's impossibility theorem and Galois theory, explicit formulae for solutions to such systems by radicals are unlikely to exist. So numerical computation arises naturally in the solution to such systems. Homotopy continuation methods represent a major class of numerical methods for solving polynomial systems.

Instead of attacking a polynomial system $P(\mathbf{x}) = \mathbf{0}$ head on, the homotopy continuation

methods consider it as a member of a family of closely related polynomial systems parametrized by a single real parameter. One member $Q(\mathbf{x}) = \mathbf{0}$ of this family should be trivial to solve. The solutions of this trivial system should be connected via smooth solution paths to all isolated solutions of the target system $P(\mathbf{x}) = \mathbf{0}$. More precisely, we construct a homotopy $H : \mathbb{C}^n \times [0, 1] \rightarrow \mathbb{C}^n$ between the given polynomial system P and some chosen system Q : H is a continuous map from the product space $\mathbb{C}^n \times [0, 1]$ to $\mathbb{C}^n$ such that $H(\mathbf{x}, 0) \equiv Q(\mathbf{x})$ and $H(\mathbf{x}, 1) \equiv P(\mathbf{x})$. Throughout this article we require $H(\mathbf{x}, t)$ to be algebraic in $\mathbf{x}$, although the method is widely applicable to nonlinear homotopies. For the purpose of the endgame analysis in Chapter 3, we also require $H(\mathbf{x}, t)$ to be holomorphic in t on a domain containing the unit interval $(0, 1) \subset \mathbb{R} \subset \mathbb{C}$ when t is considered as a complex variable. We further require the construction of $H(\mathbf{x}, t)$ to have the following three properties [17]:

1. (Triviality) The solutions of $H(\mathbf{x}, 0) = Q(\mathbf{x}) = \mathbf{0}$ are known.
2. (Smoothness) The solution set of $H(\mathbf{x}, t) = \mathbf{0}$ for $t \in (0, 1)$ consists of a finite number of smooth paths in $\mathbb{C}^n \times (0, 1)$, each parametrized by t .
3. (Accessibility) Every isolated solution of $H(\mathbf{x}, 1) = P(\mathbf{x}) = \mathbf{0}$ is reached by some path originating from $t = 0$, at a solution of $H(\mathbf{x}, 0) = Q(\mathbf{x}) = \mathbf{0}$.

When the three properties hold, the solution paths defined by $H(\mathbf{x}, t) = \mathbf{0}$ can be traced from the initial points, the solutions of $Q(\mathbf{x}) = \mathbf{0}$, at $t = 0$ to all solutions of the target problem $P(\mathbf{x}) = \mathbf{0}$ using standard numerical techniques. During the last three decades, homotopy continuation methods has seen extremely active development and grown into a large class containing a wide range of different homotopy constructions including the Total Degree Homotopy [16], the m -Homogeneous Homotopy [22], the Random Product Homotopy [18], the Cheater's Homotopy [19], the Polyhedral Homotopy [10], and many more. The homotopy

continuation method has been proved to be a reliable and efficient class of numerical methods for approximating all isolated zeros of polynomial systems. Furthermore, as it matured, it is now used as the basic building block for other numerical methods, such as numerical irreducible decomposition algorithms [25], opening up new possibilities.

1.2 Basic affine path tracking algorithms

Fix any path $\gamma \subset \mathbb{C}^n \times (0, 1)$ defined by the homotopy $H(\mathbf{x}, t) = \mathbf{0}$, by the smoothness assumption, γ can be parametrized by the t -variable, and we can write $\mathbf{x}$ as a smooth function $\mathbf{x}(t)$ of t with $H(\mathbf{x}(t), t) = \mathbf{0}$. Then it is easy to see that the function $\mathbf{x}(t)$ must satisfy the system of ordinary differential equation

$$H_{\mathbf{x}}(\mathbf{x}(t), t) \cdot \frac{d\mathbf{x}}{dt} + H_t(\mathbf{x}(t), t) = \mathbf{0}, \quad (1.1)$$

or simply $H_{\mathbf{x}} \frac{d\mathbf{x}}{dt} = -H_t$, commonly known as the DAVIDENKO differential equation [1]. This forms the basis of the numerical path tracking algorithms with which one can trace a solution path from its starting point. While any numerical solver of ordinary differential equations can, in principle, be applied to Equation (1.1) and thus used for path tracking, the special class of *predictor-corrector* method is generally preferred. In this scheme, an efficient but potentially inaccurate “predictor” is responsible for producing a rough estimate of the next point on the path using the information of known points on the path. Then a series of Newton-like “corrector” iterations is employed to bring the point approximately back to the path.

One of the most basic predictor-corrector configuration is the duet of Euler’s method and Newton’s iteration in which the prediction $\tilde{\mathbf{x}}(t_0 + \Delta t)$ for the value of $\mathbf{x}$ at $t_1 = t_0 + \Delta t$ is

given by

$$\tilde{\mathbf{x}}(t_0 + \Delta t) = \mathbf{x}(t_0) + \Delta t \cdot \dot{\mathbf{x}}(t_0) = \mathbf{x}(t_0) - \Delta t \cdot H_{\mathbf{x}}^{-1}(\mathbf{x}(t_0), t_0) \cdot H_t(\mathbf{x}(t_0), t_0).$$

This prediction step is followed by a series of Newton's iteration: at $t_1 = t_0 + \Delta t$, the equation $H(\mathbf{x}, t_1) = \mathbf{0}$ becomes a system of n equations in n unknowns. So Newton's iterations can be used to refine the prediction $\tilde{\mathbf{x}}(t_1)$ with

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left[H_{\mathbf{x}}(\mathbf{x}^k, t_1) \right]^{-1} H(\mathbf{x}^k, t_1)$$

for $k = 0, 1 \dots$, where $\mathbf{x}^0 = \tilde{\mathbf{x}}(t_1)$ is the start point.

Higher order predictors such as Hermite interpolation, Adam-Bashforth methods, and Runge-Kutta methods may be used to provide better predictions. More sophisticated correctors which can improve the convergence property of Newton's method also exist. It is the goal of this article to extend the path tracking algorithms to the complex projective space.

1.3 Numerical condition of paths

In the general context of numerical computation, it is a fair question to ask: What does a numerical algorithm do? While it is generally assumed that numerical algorithms find approximate solutions of a given problem, it was pointed out by WILKINSON [31] that they actually compute exact solutions to a nearby problem. The distance between the two problems is known as the *backward error*. Whether or not this solution is close to the solution of the original problem depends, greatly, on the sensitivity of a solution under certain perturbation

of the problem. The *condition number* is a numerical measurement of this kind of sensitivity. For instance, in the case of a system of linear equations of the form $A\mathbf{x} = \mathbf{b}$, the condition number, respect to some matrix norm $\|\bullet\|$ is given by $\text{cond}(A) = \|A\| \cdot \|A^{-1}\|$ or ∞ when A is singular. Simply put, the condition number and errors are related by the inequality

$$\text{Forward error} \leq (\text{Condition number}) \cdot (\text{Backward error}). \quad (1.2)$$

We refer to [26] for the more precise statements. It is important to note that while the backward error is controlled by the algorithm and computing devices used, the condition number is a property of the problem formulation itself. When the condition number is sufficiently large, the problem is said to be ill-conditioned, and according to inequality (1.2), one cannot provably control the forward error whenever any backward error is present. In other words, *when the question is bad, there is no good answer.*

Here we would like to assign such a condition number to the path tracking problems homotopy continuation methods intend to solve. We found it unlikely that a single number can characterize the condition of such a complex problem. In this section, we will define a weaker concept, the *condition of a homotopy path at a point*, in terms of a specific linear equation. We shall then justify the usefulness of such a concept.

Fix a path $\gamma \subset \mathbb{C}^{n+1} \times (0, 1)$ defined by $H(\mathbf{x}, t) = \mathbf{0}$. By the smoothness condition of the homotopy construction, γ can be parametrized by t as a smooth function $\mathbf{x}(t)$ on $(0, 1)$. We have stated in Equation (1.1) that this smooth function must satisfy the equation $H_{\mathbf{x}} \frac{d\mathbf{x}}{dt} = -H_t$. Thus, the tangent vector $\mathbf{v}$ of the smooth function $\mathbf{x}(t)$ at a fixed point $(\mathbf{x}, t)$

on the path is a solution to the linear system

$$H_{\mathbf{x}}\mathbf{v} = -H_t. \quad (1.3)$$

We will call this linear system the **tangent vector problem** and define the **condition number of the path γ at the point $(\mathbf{x}, t)$** to be the condition number of the matrix $H_{\mathbf{x}}$ on the left hand side of the above equation. If a path has a sufficiently large condition number at a point respect to some given threshold, we say the path is **ill-conditioned** at that point. In general a path is said to be **ill-conditioned** if it is ill-conditioned at any point, otherwise it is said to be **well-conditioned**. The threshold, of course, depends on many factors such as the precision of the floating point arithmetic, the desired accuracy for solutions, and the nature of the problem or its application.

1.3.1 Why path condition matters

It is worth reiterating that the path condition at a point, as defined above, is a property of the tangent vector problem (1.3) at that point determined by the homotopy used, and it is independent from the algorithm with which Equation (1.3) is solved or the precision of the floating point arithmetic in which computation is carried out. Also note that error is almost always present: Even if exact arithmetic can be used, discretization error still exists.

A direct justification to the meaning of the condition number of (1.3) is that, as an experimentally verified rule of thumb, if the condition number is 10^d , then Gaussian elimination will generally loose d decimal places worth of accuracy to rounding error [27, p.269]. Thus it is a direct measure of the sensitivity of the solution to the tangent vector problem respect to perturbation when the Gaussian elimination is used. Its effect on other methods varies.

While a large path condition number does not make the path impossible to track, it will adversely affect any path tracking method that depends on the direct or indirect solution to Equation (1.3).

From users' point of view, the effect of the path condition is twofold. First, it is a general experience that bad (large) path condition leads to very slow convergence for many numerical algorithms used for path tracking and vice versa. A quantitative discussion of the computational complexity of the path tracking in relation to the condition number can be found in [4]. Second, when the path condition number is sufficiently large, one cannot obtain approximations of the path tangent vector with any reasonable accuracy (using any algorithm in any arithmetic). This should definitely cast doubts on the validity of the final solutions obtained by the overall homotopy continuation method. In short, the tracking of ill-conditioned paths is slower and less trustworthy.

1.3.2 Ill-conditioned paths

A path can become ill-conditioned for a number of reasons. Here we will discuss four of the major causes. There are also other numerical causes and we will have more detailed analysis in later sections.

1.3.2.1 Divergent paths

By the smoothness condition, along each path we can consider $\mathbf{x}$ as a smooth function $\mathbf{x}(t)$ of t for $t \in (0, 1)$. While the hope is $\mathbf{x}(t)$ converges to a solution of the target system $P(\mathbf{x}) = 0$ in $\mathbb{C}^n$ as $t \rightarrow 1$, it could happen that $\mathbf{x}(t)$ does not converge to any point in $\mathbb{C}^n$ and instead $\|\mathbf{x}(t)\|$ becomes unbounded as $t \rightarrow 1$. In this case, we say the path $\mathbf{x}(t)$ **diverges**. Such a path is called a **divergent path**. Divergent paths have infinite arc length

and are ill-conditioned. Tracking such paths directly is generally impossible. Since the genesis of general homotopy continuation methods for solving polynomial systems, much effort has been devoted for constructing the homotopy so that the number of divergent paths is minimized. Despite the tremendous progress made in recent years, the handling of divergent paths remains an important problem.

1.3.2.2 Paths with large coordinates

Even if a path does not diverge, it is still possible for it to contain points with arbitrarily large coordinates since points in $\mathbb{C}^n \times [0, 1]$ in general have unbounded coordinates. Notice that since H is algebraic in $\mathbf{x}$, the magnitude of all non-constant entries of $H_{\mathbf{x}}$ grows unboundedly as $\|\mathbf{x}\| \rightarrow \infty$. In this situation, if entries of $H_{\mathbf{x}}$ grows at different rate, then it will likely cause the path to be ill-conditioned when $\|\mathbf{x}\|$ is large. Geometrically speaking, as the path $\mathbf{x}(t)$ escapes $\mathbb{C}^n \times [0, 1]$ the norm of its tangent vector $\dot{\mathbf{x}}(t) = \frac{d\mathbf{x}}{dt}$ will grow unboundedly forcing $H_{\mathbf{x}}$ to become closer and closer to being singular.

1.3.2.3 Paths approaching singular points in the middle

In this context, a singular point of a path is a point at which the smoothness condition fails. More precisely, it is a point where the path cannot be parametrized smoothly by the variable t . By assumption, when the homotopy is properly constructed, such points should never exist on the path away from the end point at $t = 1$. However, it is possible for a path to have a close encounter with a singularity. When the path is close to a singularity, the path condition number can also be large.

1.3.2.4 Paths approaching singular end points

The smoothness condition requires that the paths to be parametrized smoothly by t for $t \in (0, 1)$. However, it is also consistent with this assumption a path hits a singular end point at $t = 1$ in the sense that $\mathbf{x}$ is not a smooth function in t at $t = 1$. The analysis of the structure of the path near such a singular end point is the topic of Chapter 3.

1.3.2.5 Summary of ill-conditioned paths

Firstly, divergent paths are caused by the *non-compactness* of $\mathbb{C}^n$ as well as the choice of the homotopy. Secondly, path with large coordinates exists largely due to the fact that points in $\mathbb{C}^n$ have *unbounded coordinates*. Thirdly, the unfortunate encounter with singularities in the middle had the bad formulation of the homotopy itself to blame. Finally, singular end points is a feature of the target system. In this article we develop the projective path tracking method that will eliminate the first three causes of ill-conditioned paths while partially deal with the last cause.

Chapter 2

Projective Path Tracking

As we mentioned in Chapter 1, from the point of view of path condition, divergent paths are ill-conditioned. Tracking such paths are generally impossible. Divergent paths exist because $\mathbb{C}^n$ is not compact as a topological space. If we replace $\mathbb{C}^n$ with a compact topological space W in which $\mathbb{C}^n$ is embedded as a dense subset, then one can show that all homotopy paths, now in $W \times [0, 1]$, must converge to points inside W at $t = 1$ and have finite arc length [17]. Such a compact space W is called a **compactification** of $\mathbb{C}^n$. One of the most commonly used compactification in the context of algebraic geometry is the complex projective space $\mathbb{CP}^n$. In this chapter, we will develop the tools for homotopy path tracking in $\mathbb{CP}^n$.

First, we will explore existing methods for path tracking in complex projective space and discuss their need for improvement. In the second portion of this chapter we will briefly outline the Riemannian geometry of the complex projective space. While most of the standard constructions and concepts can be found in sufficiently advanced textbooks on Riemannian geometry and complex geometry, for completeness, we shall define all the concepts that are likely beyond an average 1st year graduate level course on differential geometry. Besides the nuts and bolts of the basic geometry, the end results of this purely theoretical discussion include three computational tools: the tangent space formula, the distance formula, and the geodesic definition on $\mathbb{CP}^n$. Guided by the theory, we will derive methods for projective path tracking in the third portion. Then in Section 2.6, through numerical analysis of the

algorithms we justify, theoretically, that the proposed projective path tracking algorithms *does not pollute path conditions*. In the sections that follow, we demonstrate some useful numerical techniques for projective path tracking. Finally we shall show, via computational results, that the geometrically motivated algorithms are also numerically sound.

2.1 Background: working in projective space

One definition for the complex projective space is the set of equivalent classes

$$\mathbb{CP}^n = \left(\mathbb{C}^{n+1} \setminus \{(0, \dots, 0)\} \right) / \sim \quad (2.1)$$

where $\mathbf{x} \sim \mathbf{y}$ for $\mathbf{x}, \mathbf{y} \in \mathbb{C}^{n+1}$ if there exists a $\lambda \in \mathbb{C} \setminus \{0\}$ such that $\mathbf{x} = \lambda \mathbf{y}$. In other words, points of $\mathbb{CP}^n$ are one dimensional linear subspaces of $\mathbb{C}^{n+1}$ with the “origin” deleted. It is common to use the notation $[x_0 : \dots : x_n]$ for the **homogeneous coordinate** of a point in $\mathbb{CP}^n$ with $[x_0 : \dots : x_n]$ being equivalent to $[\lambda x_0 : \dots : \lambda x_n]$ for any $\lambda \in \mathbb{C} \setminus \{0\}$. With such coordinates $\mathbb{CP}^n$, as a set, can be covered by subsets $U_j = \{[x_0 : \dots : x_n] \mid x_j \neq 0\}$ for $j = 0, \dots, n$, called **standard charts**. Clearly, each standard chart U_j is equivalent to $\mathbb{C}^n$, as a set, via the identification $[x_0 : \dots : x_n] \mapsto \left(\frac{x_0}{x_j}, \dots, \frac{x_{j-1}}{x_j}, \frac{x_{j+1}}{x_j}, \dots, \frac{x_n}{x_j} \right)$. These charts equip the set $\mathbb{CP}^n$ the structure of a complex manifold [28].

The zero sets of polynomials in $\mathbb{CP}^n$ are not well defined in general since each point in $\mathbb{CP}^n$ has infinitely many different coordinates. However, the zero set of a homogeneous polynomial $p \in \mathbb{C}[x_0, \dots, x_n]$ is indeed well defined. More generally, for a system $P = (p_1, \dots, p_n)$ of homogeneous polynomials, the common zero set of P in $\mathbb{CP}^n$ is also well defined. So given

any polynomial $f \in \mathbb{C}[x_1, \dots, x_n]$ of degree d , we can define its **homogenization** to be

$$\hat{f}(x_0, \dots, x_n) = x_0^d f\left(\frac{x_1}{x_0}, \dots, \frac{x_n}{x_0}\right)$$

which is clearly homogeneous of degree d , i.e.,

$$\hat{f}(\lambda x_0, \dots, \lambda x_n) = \lambda^d \hat{f}(x_0, \dots, x_n)$$

so its zero set is well defined in $\mathbb{CP}^n$. Yet $\hat{f}$ is still closely related to f in the sense that $\hat{f}(1, x_1, \dots, x_n) = f(x_1, \dots, x_n)$, i.e., $\hat{f}$ and f are equivalent on the chart U_0 defined by $x_0 \neq 0$. This common construction allows us to “lift” a problem into the complex projective space. Detailed discussion about this technique can be found in any textbook on basic algebraic geometry.

To apply this to the homotopy continuation method, given a homotopy $H(x_1, \dots, x_n, t) = (h_1, \dots, h_n)$ defined on $\mathbb{C}^n \times [0, 1]$ that is algebraic in the variables $x_1, \dots, x_n$, we shall consider their homogenizations respect to $(x_1, \dots, x_n)$

$$\hat{h}_j(x_0, \dots, x_n, t) = x_0^{d_j} h_j\left(\frac{x_1}{x_0}, \dots, \frac{x_n}{x_0}, t\right)$$

for each $j = 1, \dots, n$ where $d_j = \deg h_j$, and the new homotopy $\hat{H}(x_0, x_1, \dots, x_n, t) = (\hat{h}_1, \dots, \hat{h}_n)$, which is now defined on $\mathbb{C}^{n+1} \times [0, 1]$. Then for any fixed $t \in [0, 1]$ the common zero set of $\hat{H}(x_0, \dots, x_n, t)$ in $\mathbb{CP}^n$ is indeed well defined. We hence consider the equation $\hat{H} = \mathbf{0}$ to define paths in $\mathbb{CP}^n \times [0, 1]$ which we shall call **projective paths**. To avoid confusion, the original paths defined by $H = \mathbf{0}$ in $\mathbb{C}^n \times [0, 1]$ will be called **affine paths**. Clearly, for any path $\gamma \subset \mathbb{C}^n \times [0, 1]$ defined by $H = \mathbf{0}$, $\hat{\gamma} = \{([1, x_1, \dots, x_n], t) \mid (x_1, \dots, x_n, t) \in \gamma\}$ give

us a projective path defined by $\hat{H} = \mathbf{0}$. In this case, we call γ the **affine associate** of $\hat{\gamma}$. One key advantage of working in $\mathbb{CP}^n$ is that it is *compact*, as a topological space, thus all projective paths defined by $\hat{H} = \mathbf{0}$ must converge and be of finite length. In the following subsections, we will discuss existing path tracking methods in the projective space $\mathbb{CP}^n$. One important theorem that is used many times in our discussion is the Euler's theorem which we shall state here for an easy reference.

Theorem 1. (Euler's identity for homogeneous functions) *Given a smooth homogeneous function $f(x_0, \dots, x_n)$ of order d ,*

$$\sum_{j=0}^n x_j \frac{\partial f}{\partial x_j} = d \cdot f(x_0, \dots, x_n).$$

[29, Theorem 10.2] contains a simple proof of this theorem. From this we can deduce the following useful fact:

Corollary 1. *Let $F = (F_1, \dots, F_n)$ with each $F_j \in \mathbb{C}[x_0, \dots, x_n]$ being homogeneous, and let $(x_0, \dots, x_n) \in \mathbb{C}^{n+1}$ be a point such that $F(x_0, \dots, x_n) = \mathbf{0}$, then*

$$DF_{(x_0, \dots, x_n)} \begin{pmatrix} x_0 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}.$$

In other words, a common zero $\mathbf{x} \in \mathbb{C}^{n+1}$ of a system of homogeneous polynomial is also a right zero vector of the Jacobian matrix of the system at $\mathbf{x}$.

2.1.1 Projective path tracking using affine charts

The major difficulty in working with $\hat{H}$, defined on $\mathbb{CP}^n \times [0, 1]$ is that for a fixed $t \in [0, 1]$ a solution to $\hat{H} = \mathbf{0}$, being a point in $\mathbb{CP}^n$, has infinitely many different but equivalent homogeneous coordinates. When it is considered as a system of equations in $\mathbb{C}^{n+1}$ for fixed t , $\hat{H} = \mathbf{0}$ has $n + 1$ unknowns but only n equations and hence has no isolated solutions. Thus traditional path tracking algorithms cannot be directly applied to the problem.

In theory, this problem can be resolved by simply transferring the problem $\hat{H} = \mathbf{0}$ back to the affine space $\mathbb{C}^n$. As we have just stated, $\mathbb{CP}^n$ is covered by standard charts $U_j = \{[x_0 : \cdots : x_n] \mid x_j \neq 0\}$ for $j = 0, \dots, n$ with each being identified with a copy of $\mathbb{C}^n$. So one could simply work in one such chart. On the chart defined by $x_j \neq 0$, after scaling, we can assume $x_j = 1$. Then on this chart $\hat{H} = \mathbf{0}$ can be simply represented by the new system of equations

$$\begin{cases} \hat{H}(x_0, \dots, x_n, t) = 0 \\ x_j - 1 = 0. \end{cases}$$

In particular, using $j = 0$ we will obtain exactly the original problem we have started with, since $\hat{H}(1, x_1, \dots, x_n, t) = H(x_1, \dots, x_n, t)$. The above equation is an equation in $\mathbb{C}^{n+1} \times [0, 1]$, and existing affine path tracking algorithms for $\mathbb{C}^{n+1}$ can be used.

This method can be further generalized. A geometric interpretation of the last equation $x_j - 1 = 0$ in the above system is that it defines an affine subspace of (complex) codimension one, i.e., a hyperplane L of $\mathbb{C}^{n+1}$. Then the above system can be interpreted as the projection of the projective homotopy paths defined by $\hat{H} = \mathbf{0}$ to the hyperplane L . With this interpretation in mind, there is no reason to restrict ourselves in using only the hyperplanes defined by $x_j - 1 = 0$. We can generalize the method by using a hyperplane L of $\mathbb{C}^{n+1}$

defined by a linear equation $a_0x_0 + \cdots + a_nx_n - 1 = 0$, i.e., we can consider the system induced by L

$$\hat{H}^L(\mathbf{x}, t) = \begin{cases} \hat{H}(x_0, \dots, x_n, t) & = 0 \\ a_0x_0 + \cdots + a_nx_n - 1 & = 0. \end{cases} \quad (2.2)$$

For later reference, we state the following lemma that was shown in [21] and [25]:

Lemma 1. *As long as the original homotopy H satisfies the smoothness condition, given any fixed $t \in (0, 1)$, for almost all L , the projection of the zero set of $\hat{H} = \mathbf{0}$ to L consists of smooth points.*

Thus one can simply choose a generic hyperplane L , and apply affine path tracking algorithms to the homotopy $\hat{H}^L = \mathbf{0}$ induced by L . Methods based on this idea are widely used and are adopted by software packages such as BERTINI and HOM4PS-3.0 with useful results.

However, this method is not completely without flaws. First of all, while $\mathbb{CP}^n$ itself is compact, its projection onto the hyperplane L in $\mathbb{C}^{n+1}$ is not. So during the path tracking process the magnitude of the components x_j for $j = 0, \dots, n$ can still grow without bound, which is indeed the original problem we set out to solve. But most importantly, for the numerical path tracking algorithms to *work well in real world*, it is crucially important that the paths not only satisfy the theoretical smoothness but also possess good numerical path condition. To illustrate the problem in this regard, let us consider an actual computational result where the homotopy method based on Equation (2.2) is used to solve the `cyclic7` [3] problem. A randomly chosen $\mathbf{a} = (a_0, \dots, a_n)$ is used to represent the generic hyperplane L . Figure 2.1 shows the path condition number along a single path.

Clearly visible in the graph is the fact that the path condition can reach close to 10^{16} . It is

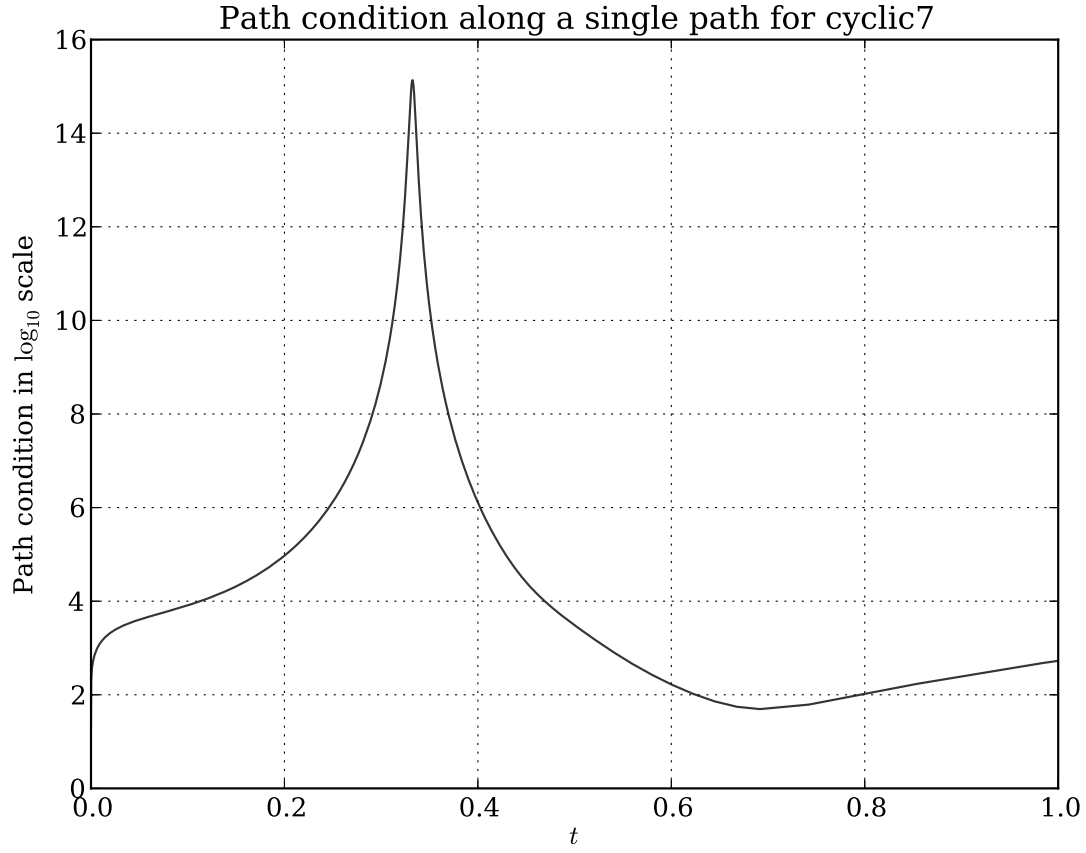


Figure 2.1: The path condition number, in $\log_{10}$ scale, of a single path defined by Equation (2.2) $\hat{H}^L = \mathbf{0}$ for solving the `cyclic7` problem: the path condition can be as large as 10^{16} “in the middle” of the path tracking.

tempting to conclude that we had a close encounter with a singularity and hence the original homotopy H or even the target system (`cyclic7`) is to be blamed. However, when compared with the path conditions of the original affine associate of exactly the same projective path, shown in Figure 2.2, the difference is quite striking: while the path condition number of the original affine associate was kept well under 100 throughout the entire path tracking process, the projection of the same projective path to L has path condition close to 10^{16} at certain points. So the projection to the hyperplane L , in this particular case, has *polluted the path condition*.

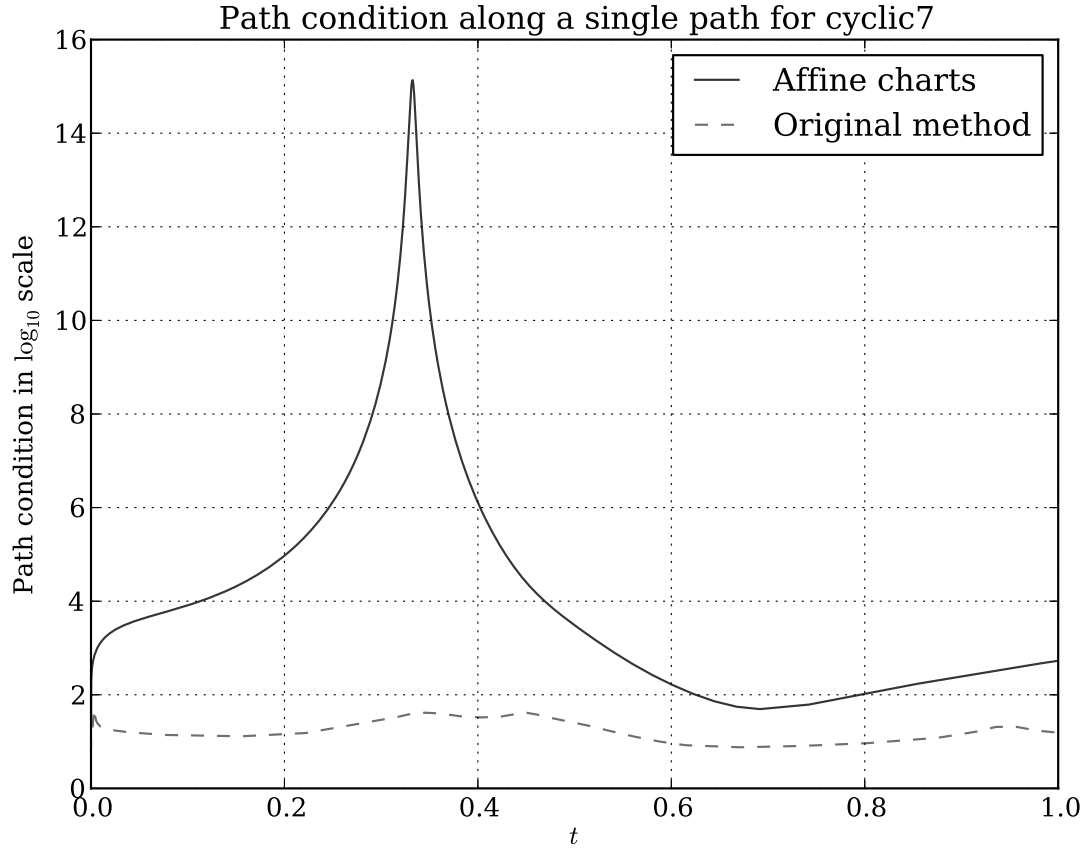


Figure 2.2: The path condition numbers, in $\log_{10}$ scale, of a path defined by Equation (2.2) $\hat{H}^L = 0$ (shown in solid lines) and that of the affine associate of the same projective path (shown in dashed line) defined by the original homotopy $H = \mathbf{0}$ tracked for solving the `cyclic7` problem. The two paths represent the same projective path in $\mathbb{CP}^n \times [0, 1]$ defined by $\hat{H} = \mathbf{0}$, but the difference in their respective path condition numbers is striking!

To understand the cause, we shall now analyze the path condition when the homotopy defined in Equation (2.2) is used.

2.1.2 Numerical analysis of the homotopy path

As we defined in Chapter 1, at a fixed $(\mathbf{x}, t) = (x_0, \dots, x_n, t) \in \mathbb{C}^{n+1} \times [0, 1]$ on a path defined by Equation (2.2), the path condition is the condition number of the Jacobian matrix

$$J = \begin{pmatrix} \frac{\partial \hat{H}_1}{\partial x_0} & \cdots & \frac{\partial \hat{H}_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial \hat{H}_n}{\partial x_0} & \cdots & \frac{\partial \hat{H}_n}{\partial x_n} \\ a_0 & \cdots & a_n \end{pmatrix}$$

of $\hat{H}^L$ respect to $\mathbf{x} = (x_0, \dots, x_n)$ evaluated at $(\mathbf{x}, t)$. In light of Corollary 1, we can see that

$$J \cdot \frac{\mathbf{x}}{\|\mathbf{x}\|_2} = \frac{1}{\|\mathbf{x}\|_2} \cdot \begin{pmatrix} 0 \\ \vdots \\ 0 \\ a_0 x_0 + \cdots + a_n x_n \end{pmatrix} = \alpha \cdot \mathbf{e}_{n+1}$$

where $\alpha := (a_0 x_0 + \cdots + a_n x_n) / \|\mathbf{x}\|_2$ and $\mathbf{e}_{n+1} = (0, \dots, 0, 1) \in \mathbb{C}^{n+1}$. By assumption $\alpha \neq 0$, hence we have $J^{-1} \mathbf{e}_{n+1} = \frac{\mathbf{x}}{\alpha \|\mathbf{x}\|_2}$, so for the unit vector $\mathbf{e}_{n+1}$,

$$\|J^{-1} \mathbf{e}_{n+1}\|_2 = \left\| \frac{\mathbf{x}}{\alpha \|\mathbf{x}\|_2} \right\|_2 = \frac{1}{|\alpha|}.$$

Therefore $\|J^{-1}\|_2 \geq \frac{1}{|\alpha|}$ and the path condition at $(\mathbf{x}, t)$ is at least $\frac{1}{|\alpha|}$. When the absolute value of α is small the path becomes ill-conditioned. Given any threshold μ for path condition the set of $\mathbf{a} = (a_0, \dots, a_n) \in \mathbb{C}^{n+1}$ that cause the path condition at $(\mathbf{x}, t)$ to be greater than μ contains at least the set $\{\mathbf{a} \in \mathbb{C}^{n+1} : |\mathbf{a}^\top \mathbf{x}| / \|\mathbf{x}\|_2 < 1/\mu\}$ even if the original homotopy

$H(\mathbf{x}, t)$ is well-conditioned. Furthermore, unless the chart is to be changed at every step, the same chart must be used on some t -intervals $[t_1, t_2]$, then the set of $\mathbf{a}$ that will cause the path condition to be greater than μ on that t -interval contains at least

$$\left\{ \mathbf{a} \in \mathbb{C}^{n+1} : |\mathbf{a}^\top \mathbf{x}(t)| / \|\mathbf{x}(t)\|_2 < 1/\mu \text{ for } t \in [t_1, t_2] \right\}$$

which could potentially be of rather large $\mathbb{C}^{n+1}$ Lebesgue measure. From a probability point of view, while for a fixed $(\mathbf{x}, t)$ on a path the choices of $\mathbf{a} \in \mathbb{C}^{n+1}$ making the path *singular* is small enough to be ignored, the choices of $\mathbf{a}$ making the path *ill-conditioned* may not be. In other words, *while “wrong” choices are rare, bad choices are plenty*. In this case it can be hard to avoid them.

To have a quantitative idea of the probability of making such bad choices, we shall look at a distribution of maximum path condition numbers. The `cyclic7` problem computation is repeated 10 times using both homotopies $H = \mathbf{0}$ and $\hat{H}^L = \mathbf{0}$. The maximum path condition along each path is collected. The distribution of maximum path condition, in $\log_{10}$ scale, resulted from the two different approach is shown in Figure 2.3. The two approaches are essentially tracking the same paths: the only difference is the hyperplanes the paths are projected to. However, the difference in the distribution of the maximum path condition is striking.

Clearly shown in the histogram is the fraction of ill-conditioned paths defined by $\hat{H}^L = \mathbf{0}$. While double-precision can still track all the paths and obtain the correct answers, in light of Inequality (1.2), floating point arithmetic with much higher precision must be used if one intend to obtain any provable accuracy. As expected, the bad condition has also made the path tracking process much slower: in this particular experiment, while the number of paths

with maximum path condition greater than 10^{10} is less than 5% among the total number of paths, the amount of time the path tracking algorithm spent on them is actually more than what it spent on the remaining 95% of the paths!

The `cyclic7` problem is not alone in revealing the numerical problem with the method based on affine charts. As we can see in Table 2.1, the same numerical problem exists in the path tracking of many systems.

System	Percentage with condition $\geq 10^{10}$	Percentage with condition $\geq 10^{15}$
<code>cyclic5</code> [3]	1.67%	0.83%
<code>cyclic7</code> [3]	4.73%	0.8%
<code>cyclic11</code> [3]	4.97%	1.03%
<code>eco11</code> [21]	0.91%	0.91%
<code>eco15</code> [21]	0.99%	0.71%
<code>9-point</code> [30]	0.99%	0.12%

Table 2.1: Average percentage, rounded to the second digit after the decimal point, of paths with large maximum path condition number over 10 different runs when the homotopy $\hat{H}^L = \mathbf{0}$ of Equation (2.2) is used to solve different polynomial systems (all divergent paths and those with singular end points are ignored). At first glance, the percentages seem rather small, however one must keep in mind that hundreds of thousands of paths are tracked in solving some of the systems, so even a very small percentage will translate to a large absolute number of paths with very high maximum path condition.

Clearly, more numerically stable method for path tracking in the complex projective space is needed. In the remaining part of this chapter, we shall develop a projective path tracking method from the point of view of the Riemannian geometry on $\mathbb{CP}^n$.

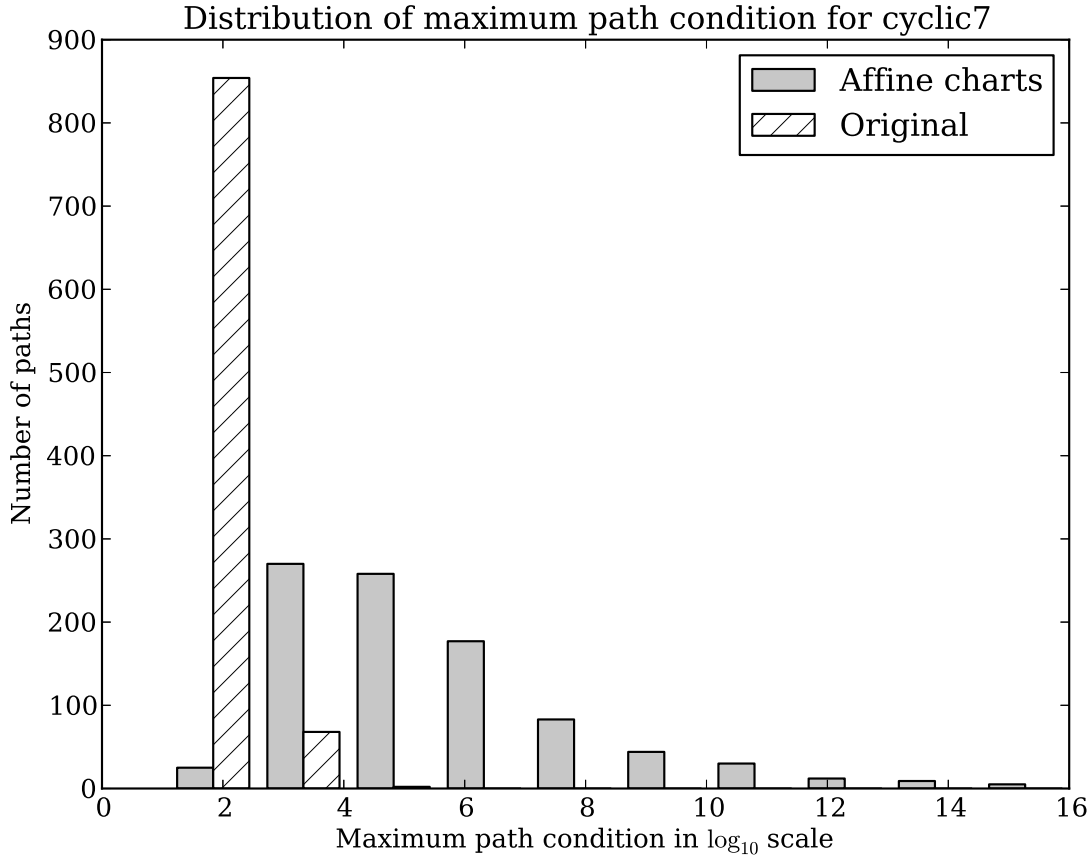


Figure 2.3: The histogram of maximum path condition in $\log_{10}$ scale for all paths defined by the original homotopy $H = \mathbf{0}$ (in striped bars) and those defined by the homotopy $\hat{H}^L = \mathbf{0}$ of Equation (2.2) (in solid bars) for solving the `cyclic7` problem. The height of each bar indicates the total number of paths having the corresponding maximum path condition. For example the leftmost striped bar has a height of 852 and occupies a horizontal interval near 2. This bar signifies that there are 852 paths defined by the original $H = \mathbf{0}$ having maximum path condition approximately in the scale of 10^2 . This graph represents the average results of 10 different runs. While the two approaches are essentially tracking the different projections of the same set of projective paths defined by $\hat{H} = \mathbf{0}$, the difference in maximum path condition distribution is indeed striking!

2.2 Geometry of the complex projective space

First, we shall review the Riemannian geometry of $\mathbb{CP}^n$. The definition $\mathbb{CP}^n = (\mathbb{C}^{n+1} \setminus \{(0, \dots, 0)\}) / \sim$ merely describes $\mathbb{CP}^n$ as a set and carries no obvious geometric structure. While the standard charts $U_0, \dots, U_n$ used above equip $\mathbb{CP}^n$ the structure of a complex manifold, here we shall take another classical route: $\mathbb{CP}^n$ can be realized as the quotient manifold of S^{2n+1} by the action of a compact Lie group, the circle group. This point of view provides us the “natural” choices for the smooth and Riemannian structure for $\mathbb{CP}^n$. With these additional geometric structures, we can derive tools for projective path tracking algorithms.

2.2.1 Identification between $\mathbb{C}$ and $\mathbb{R}^2$

We identify $\mathbb{C}$ with $\mathbb{R}^2$ via the map $x + iy \mapsto (x, y)^\top$. Under such an identification, the multiplication by a complex number $a + bi$ can be considered as an invertible linear transformation on $\mathbb{R}^2$. It is easy to check that it is represented by the 2×2 real valued matrix

$$\begin{pmatrix} a & -b \\ b & a \end{pmatrix}.$$

We can of course extend this and identify $\mathbb{C}^m$ and $\mathbb{R}^{2m}$ via $(z_1, \dots, z_m) \mapsto (\operatorname{Re} z_1, \operatorname{Im} z_1, \dots, \operatorname{Re} z_m, \operatorname{Im} z_m)$. Then the scalar multiplication to vectors in $\mathbb{C}^m$ by the complex number $a + bi$, under such an identification, can be viewed as the linear transformation of $\mathbb{R}^{2m}$ given

by the $2m \times 2m$ matrix

$$\begin{pmatrix} a & -b & & & \\ b & a & & & \\ & & \ddots & & \\ & & & a & -b \\ & & & b & a \end{pmatrix}.$$

In the context of complex geometry, it can be immediately recognized that we are simply describing the standard complex structure on $\mathbb{R}^{2m}$.

One such scalar multiplication is of special interest to us: multiplying by a complex number $e^{i\theta} = \cos \theta + \sin \theta i$ on the unit circle $U = \{u \in \mathbb{C} : |u| = 1\}$ of $\mathbb{C}$, which preserves length. As a linear transformation on $\mathbb{R}^{2m}$, we shall use the same notation

$$e^{i\theta} = \begin{pmatrix} \cos \theta & -\sin \theta & & & \\ \sin \theta & \cos \theta & & & \\ & & \ddots & & \\ & & & \cos \theta & -\sin \theta \\ & & & \sin \theta & \cos \theta \end{pmatrix} \quad (2.3)$$

to represent the block diagonal matrix on the right hand side, which is clearly orthogonal. One can easily check the familiar statements $(e^{i\theta})^{-1} = e^{i(-\theta)}$ and $e^{i\theta_1} \cdot e^{i\theta_2} = e^{i(\theta_1+\theta_2)}$ are still true as statements of linear transformations, and that $1 = e^{i0}$ is the identity matrix. Simply put, there is a natural isomorphism between the multiplicative group of the unit circle U of $\mathbb{C}$ and that of the orthogonal matrices of the above form. To go one step further, for a

fixed point $\mathbf{v} \in \mathbb{R}^{2n}$ the statement

$$\frac{d}{d\theta} e^{i\theta} \mathbf{v} = \begin{pmatrix} -\sin \theta & -\cos \theta & & & \\ \cos \theta & -\sin \theta & & & \\ & & \ddots & & \\ & & & -\sin \theta & -\cos \theta \\ & & & \cos \theta & -\sin \theta \end{pmatrix} \cdot \mathbf{v} = ie^{i\theta} \mathbf{v}$$

still holds in this context. For $\mathbf{u}, \mathbf{v} \in \mathbb{R}^{2n}$, the notation

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbb{R}} = \mathbf{u}^{\top} \mathbf{v}$$

always denotes the standard dot product in the Euclidean space. We will use the same symbols $\mathbf{u}$ and $\mathbf{v}$ for the two corresponding vectors in $\mathbb{C}^n$ and the notation

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbb{C}} := \mathbf{u}^H \mathbf{v}$$

for the complex inner product, where $\mathbf{u}^H$ is the conjugate transpose of the complex vector $\mathbf{u} \in \mathbb{C}^n$. There are other possible definitions for the complex inner product. This one is chosen so that the two inner products have an obvious connection:

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbb{R}} = \operatorname{Re} \langle \mathbf{u}, \mathbf{v} \rangle_{\mathbb{C}} \quad \text{and} \quad \langle i\mathbf{u}, \mathbf{v} \rangle_{\mathbb{R}} = \operatorname{Im} \langle \mathbf{u}, \mathbf{v} \rangle_{\mathbb{C}} . \quad (2.4)$$

It is also a convenient fact that the length of a vector $\|\mathbf{u}\|_2 = \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle_{\mathbb{R}}} = \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle_{\mathbb{C}}}$ regardless which inner product is used.

2.2.2 Notation and terminology

In the rest of this chapter the term **smooth** always means infinitely differentiable while a **manifold** is always a connected smooth manifold that is Hausdorff and second-countable. A **smooth curve** is by assumption a smoothly parametrized smooth submanifold of (real) dimension 1. At a point $\mathbf{x}$ of a manifold M , we use the notation $T_{\mathbf{x}}M$ for the tangent space of M at $\mathbf{x}$. The tangent bundle of M is denoted by TM while its smooth vector fields is denoted by $\mathfrak{X}(M)$. When considering certain group action on M , we use $[\mathbf{x}]$ for the orbit of $\mathbf{x} \in M$ under that action.

Since both inner products $\langle \bullet, \bullet \rangle_{\mathbb{R}}$ and $\langle \bullet, \bullet \rangle_{\mathbb{C}}$ will come into our discussion, we must use the concept of *orthogonality* with care. The word **orthogonal** should always mean the relationship that two (real) vectors have their real inner product being zero. The term **orthogonal compliment** should carry the same meaning in real.

2.2.3 Realizing $\mathbb{CP}^n$ as S^{2n+1}/U

Let $S^{2n+1} = \{\mathbf{x} \in \mathbb{R}^{2n+2} : \|\mathbf{x}\|_2 = 1\} = \{\mathbf{z} \in \mathbb{C}^{n+1} : \|\mathbf{z}\|_2 = 1\}$ be the $(2n+1)$ -dimensional sphere. It is a standard construction to view $\mathbb{CP}^n$ as the quotient of S^{2n+1} under a group action. This construction will provide us the smooth and Riemannian structure for $\mathbb{CP}^n$.

It is clear that each point $(z_0, \dots, z_n) \in S^{2n+1}$ represents a point $[z_0 : \dots : z_n]$ in $\mathbb{CP}^n$, and each point in $\mathbb{CP}^n$ has such a representative. More formally, let $\pi : S^{2n+1} \rightarrow \mathbb{CP}^n$ be the map given by $(z_0, \dots, z_n) \mapsto [z_0 : \dots : z_n]$, then π is clearly onto. We would like to use S^{2n+1} as our model for $\mathbb{CP}^n$ via the map π . It has the obvious advantage that all coordinates for all points are bounded. Indeed $\|\mathbf{x}\|_2 = 1$ for all $\mathbf{x} \in S^{2n+1}$. However, the representative of a point in $\mathbb{CP}^n$ is not unique, i.e., π is not 1-to-1, as $\pi(\mathbf{x}) = \pi(\lambda\mathbf{x})$ for any

$\lambda \in \mathbb{C}^* = \mathbb{C} \setminus \{0\}$. But to leave S^{2n+1} invariant, we must have $\lambda = e^{i\theta}$. The multiplication by $e^{i\theta}$, viewed as a transformation on $\mathbb{R}^{2n+2}$, is given by the orthogonal matrix shown in Equation (2.3). It is easy to check that for $\mathbf{x} \in S^{2n+1}$, the points of the form $e^{i\theta}\mathbf{x}$ with $\theta \in \mathbb{R}$ are exactly those that represent the same point as $\mathbf{x}$ itself. Formally speaking,

$$\pi^{-1}(\pi(\mathbf{x})) = \{e^{i\theta}\mathbf{x} \mid \theta \in \mathbb{R}\}.$$

Therefore, as a set, we can identify $\mathbb{CP}^n$ with the set of equivalent classes $\{[\mathbf{x}] : \mathbf{x} \in S^{2n+1}\}$ where

$$[\mathbf{x}] := \{e^{i\theta}\mathbf{x} \mid \theta \in \mathbb{R}\}.$$

Next we shall show that this identification is also geometric. Let $U = \{e^{i\theta} \mid \theta \in \mathbb{R}\}$ be the unit circle which is a compact smooth submanifold of $\mathbb{C} \approx \mathbb{R}^2$. U clearly forms a group under multiplication, which is itself a smooth operation. In other words, U is a compact Lie group. From this point of view each orbit $U\mathbf{x} = [\mathbf{x}]$ is exactly the set of points representing the same point in $\mathbb{CP}^n$ via the map π . Namely, we can consider $\mathbb{CP}^n$ as the quotient S^{2n+1}/U of S^{2n+1} under the group action of U . It is easy to check that U acts on S^{2n+1} smoothly and freely. Because $U \subset \mathbb{C}$ is compact, the action is also proper. By the Quotient Manifold Theorem [13, Theorem 9.16] $\mathbb{CP}^n = S^{2n+1}/U$ is a smooth manifold in its own right, and it has the unique smooth structure such that π is a smooth submersion, and at each point $\mathbf{x} \in S^{2n+1}$ we have the decomposition

$$T_{\mathbf{x}}S^{2n+1} = T_{[\mathbf{x}]} \mathbb{CP}^n \oplus T_{\mathbf{x}}[\mathbf{x}]$$

of the tangent space. Note, however, that the decomposition alone does not explicitly tell us how to choose $T_{[\mathbf{x}]} \mathbb{CP}^n$, since any complement of $T_{\mathbf{x}}[\mathbf{x}]$ in $T_{\mathbf{x}}S^{2n+1}$ is algebraically isomorphic to $T_{[\mathbf{x}]} \mathbb{CP}^n$. Nonetheless, if we take the Riemannian structure of S^{2n+1} into account, then there is essentially only one “natural” choice of $T_{[\mathbf{x}]} \mathbb{CP}^n$ and one “natural” Riemann structure on $\mathbb{CP}^n$. We shall make this statement precise.

As a submanifold of $\mathbb{R}^{2n+2}$, S^{2n+1} inherits a natural **Riemannian metric** $g_{S^{2n+1}}$, which is a smooth assignment of inner products in the tangent bundle TS^{2n+1} . Or more formally, $g_{S^{2n+1}}$ is a 2-tensor field that is symmetric and positive definite simply given by

$$g_{S^{2n+1}}(\mathbf{u}, \mathbf{v}) = \langle \mathbf{u}, \mathbf{v} \rangle_{\mathbb{R}}$$

for $\mathbf{u}, \mathbf{v} \in T_{\mathbf{x}}S^{2n+1}$ at any $\mathbf{x} \in S^{2n+1}$. Using this metric, we can decompose the tangent space at $\mathbf{x}$ into

$$T_{\mathbf{x}}S^{2n+1} = V_{\mathbf{x}} \oplus H_{\mathbf{x}}$$

where

$$V_{\mathbf{x}} = T_{\mathbf{x}}[\mathbf{x}] \quad H_{\mathbf{x}} = \{\mathbf{h} \in T_{\mathbf{x}}S^{2n+1} \mid g_{S^{2n+1}}(\mathbf{h}, \mathbf{v}) = 0, \forall \mathbf{v} \in V_{\mathbf{x}}\}.$$

That is, we can decompose $T_{\mathbf{x}}S^{2n+1}$ into the direct sum of two subspaces that are orthogonal respect to the inner product given by $g_{S^{2n+1}}$ at $\mathbf{x}$. In this case, we can simply define $T_{[\mathbf{x}]} \mathbb{CP}^n := H_{\mathbf{x}}$, the so called **horizontal space** at $\mathbf{x}$ respect to the action of U . As stated above, the canonical projection $\pi : S^{2n+1} \rightarrow S^{2n+1}/U \approx \mathbb{CP}^n$ is a smooth submersion in terms of the smooth structure given by the Quotient Manifold Theorem, i.e., at any

$\mathbf{x} \in S^{2n+1}$ the pushforward $\pi_* : T_{\mathbf{x}}S^{2n+1} \rightarrow T_{[\mathbf{x}]} \mathbb{CP}^n$ is surjective, as a linear map, at each point $\mathbf{x}$. There is a *unique* Riemannian metric $g := g_{\mathbb{CP}^n}$ for $\mathbb{CP}^n$, with respect to which π is a **Riemannian submersion**: a smooth submersion that is also an isometry on the horizontal space $H_{\mathbf{x}}$ over each point $\mathbf{x}$, i.e.,

$$g(\pi_*(h_1), \pi_*(h_2)) = g_{S^{2n+1}}(h_1, h_2) \quad \text{for all } h_1, h_2 \in H_{\mathbf{x}},$$

where π_* is the pushforward of the smooth map π . In the rest of this chapter the notation $\mathbb{CP}^n$ should always mean the n -dimensional complex projective space realized as the quotient manifold S^{2n+1}/U carrying the smooth structure determined by the quotient and the Riemannian metric g .

Actually, $\mathbb{CP}^n$ has much richer geometric structures than we shall make use of in the following discussion. The Hermitian structure (a continuous assignment of Hermitian product on the tangent bundle $T\mathbb{CP}^n$) and the complex structure, to name a few, are not explicitly used here but are naturally quite important in the study of $\mathbb{CP}^n$. We refer to [4] for a more detailed discussion.

2.2.4 Tangent formula for $\mathbb{CP}^n$

Following the notation we introduced above, at any point $\mathbf{x} \in S^{2n+1}$, the tangent space $T_{[\mathbf{x}]} \mathbb{CP}^n = H_{\mathbf{x}}$, viewed as the horizontal space of $T_{\mathbf{x}} S^{2n+1}$ respect to the action of U , consists of all the vectors that are orthogonal to the vertical space $V_{\mathbf{x}} = T_{\mathbf{x}} [\mathbf{x}]$. At $\mathbf{x}$ the tangent space $T_{\mathbf{x}} S^{2n+1}$ of S^{2n+1} , as an embedded submanifold of $\mathbb{R}^{2n+2}$, is simply the linear subspace $\{\mathbf{x}\}^\perp \subseteq T_{\mathbf{x}} \mathbb{R}^{2n+2} \cong \mathbb{R}^{2n+2}$. For $T_{\mathbf{x}} [\mathbf{x}]$, it is clear that near $\mathbf{x}$, the one (real) dimensional submanifold $[\mathbf{x}]$ is parametrized by $\gamma(\theta) = e^{i\theta} \mathbf{x}$ with $\gamma(0) = \mathbf{x}$. So

$$\dot{\gamma}(0) = ie^{i0} \mathbf{x} = i\mathbf{x}.$$

As a vector in $T_{\mathbf{x}} \mathbb{R}^{2n+2}$, it is the generator of the one dimensional vector space $T_{\mathbf{x}} [\mathbf{x}]$. So far, we have

$$T_{\mathbf{x}} [\mathbf{x}] = \text{span} \{i\mathbf{x}\} \quad \text{and} \quad T_{\mathbf{x}} S^{2n+1} = \{\mathbf{x}\}^\perp.$$

As we just stated, $T_{[\mathbf{x}]} \mathbb{CP}^n$ is the linear subspace of $T_{\mathbf{x}} S^{2n+1} = \{\mathbf{x}\}^\perp$ that is orthogonal to $T_{\mathbf{x}} [\mathbf{x}] = \{i\mathbf{x}\}$. So it is simply $\{i\mathbf{x}\}^\perp \cap \{\mathbf{x}\}^\perp$ in $T_{\mathbf{x}} \mathbb{R}^{2n+2} \cong \mathbb{R}^{2n+2}$, i.e., it is the set of vector $\mathbf{v}$ such that

$$\langle i\mathbf{x}, \mathbf{v} \rangle_{\mathbb{R}} = 0$$

$$\langle \mathbf{x}, \mathbf{v} \rangle_{\mathbb{R}} = 0$$

which is equivalent to the complex equation $\langle \mathbf{x}, \mathbf{v} \rangle_{\mathbb{C}} = 0$ based on the observation from Equation (2.4). Therefore the tangent space can be characterized as

$$T_{[\mathbf{x}]} \mathbb{CP}^n = \left\{ \mathbf{v} \in T_{\mathbf{x}} \mathbb{R}^{2n+2} : \langle \mathbf{x}, \mathbf{v} \rangle_{\mathbb{C}} = 0 \right\}.$$

In particular, for a smooth curve $\mathbf{x}(t) : [0, 1] \rightarrow \mathbb{CP}^n$, its tangent vector $\dot{\mathbf{x}}(t) \in T_{[\mathbf{x}]} \mathbb{CP}^n$ has the property that

$$\langle \mathbf{x}(t), \dot{\mathbf{x}}(t) \rangle_{\mathbb{C}} = 0$$

when both vectors are considered as vectors in $\mathbb{C}^{n+1}$. Notice that when we restrict ourselves to the point of view of Riemannian geometry we must consider the above equation as a system of two equations $\langle \mathbf{x}(t), \dot{\mathbf{x}}(t) \rangle_{\mathbb{R}} = 0$ and $\langle i\mathbf{x}(t), \dot{\mathbf{x}}(t) \rangle_{\mathbb{R}} = 0$ in $\mathbb{R}^{2n+2}$. The first one states that $\dot{\mathbf{x}}$ must lie flat in the tangent space of S^{2n+1} , while the second one requires the tangent vector to be inside the horizontal space that is orthogonal to the orbit of $\mathbf{x}$ under the group action of U . The same equation can also be derived from the point of view of the Hermitian structure on $\mathbb{CP}^n$, for that we again refer to [4].

2.2.5 Distance formula for $\mathbb{CP}^n$

The concept of distance on $\mathbb{CP}^n$ plays a number of important roles in the numerical algorithms that we will discuss later in this article, ranging from detecting the convergence of projective Newton's iterations to testing the sameness of end points in projective Cauchy integral method. A “natural” concept of distance can be derived from the Riemannian metric on $\mathbb{CP}^n$. Recall that the Riemannian metric g of a Riemannian manifold M is a smooth assignment of inner product in the tangent bundle TM . It also induces a smooth assignment ℓ of lengths to vectors in TM (a Finsler structure) given by

$$\ell(\mathbf{v}) = \sqrt{g(\mathbf{v}, \mathbf{v})}.$$

With this, for a smooth curve $\gamma : [0, 1] \rightarrow M$, we can measure its length by

$$\int_0^1 \ell(\dot{\gamma}(t)) dt$$

which is indeed invariant under reparametrization [13, Proposition 11.15] and hence a property of the curve itself. So it is meaningful to assign a length $\ell(\gamma)$ to the smooth curve γ as defined above. This definition can be easily extended to piecewise smooth curves by summing the length of individual segments. Armed with the tool for measuring length of piecewise smooth curves, we can then define the distance between two points in $\mathbb{CP}^n$ to be the infimum of the length among all piecewise smooth curves connecting the two. While quite elegant, this definition does not offer a computable formula. A more direct approach requires the concept of *geodesics*.

2.2.5.1 Riemannian geodesics

In the search of the shortest paths between two points, a.k.a. the **minimizing curves**, our intuition would be of great help. In the Euclidean space, the shortest path between two points is simply the straight line segment joining the two which also happen to be the most straight path. This concept of “most straight paths” can be generalized to Riemannian manifolds, and this generalization is known as *geodesics*. The link between “straightness” and “shortness” is preserved to a certain extent: as we will state later, all minimizing curves are geodesics as long as they are parametrized in the “right way”, and all geodesics are locally minimizing.

Intuitively, geodesics are curves that are “as straight as possible”, much like the straight line segments in the Euclidean space. The “straightness” of straight lines can be characterized precisely: a curve with constant speed parametrization is straight if and only if its acceleration

is always zero. We simply have to generalize this notion of zero acceleration to curves in Riemannian manifold. While it is tempting to define the acceleration of a curve $\gamma : [0, 1] \rightarrow M$ on a manifold M to be the second derivative “ $\ddot{\gamma}$ ”, one immediately runs into trouble with this approach: the tangent vectors at different points actually live in different tangent spaces. So to define the concept of acceleration, we need a geometrically meaningful way to capture the notion of the acceleration “ $\ddot{\gamma} = \frac{d}{dt}\dot{\gamma}$ ” along the curve γ . More formally, we need an operator that gives us a coordinate independent method for differentiating a vector field along a curve. Being the analog of $\frac{d}{dt}$, it obviously has to be linear (over $\mathbb{R}$) and satisfies the product rule. However, these requirements do not uniquely determine our desired concept of “acceleration”. Fortunately, there is an important theorem, lies deep in the heart of Riemannian geometry, which basically states that there is one “right” definition, with which we have the link between curves with zero acceleration and those with shortest length. In the following paragraphs, we shall state these ideas rigorously.

A **linear connection** [12, Lemma 4.9] on M is a bilinear map $\nabla : \mathfrak{X}(M) \times \mathfrak{X}(M) \rightarrow \mathfrak{X}(M)$, with the notation $\nabla_X Y := \nabla(X, Y)$, such that for $f_1, f_2 \in C^\infty(M)$ we have

$$\nabla_{f_1 X_1 + f_2 X_2} Y = f_1 \nabla_{X_1} Y + f_2 \nabla_{X_2} Y$$

$$\nabla_X(f_1 Y) = f_1 \nabla_X Y + (X f_1) Y.$$

We also say ∇ is **symmetric** if it satisfies $\nabla_X Y - \nabla_Y X = [X, Y]$, where $[X, Y]$ is the Lie bracket of the vector fields X and Y . Let $\mathfrak{X}(\gamma)$ be the space of all vector fields along a smooth curve γ parametrized by variable t , then respect to a given linear connection ∇ , there exists a unique linear operator [12, Lemma 4.9] $D_t : \mathfrak{X}(\gamma) \rightarrow \mathfrak{X}(\gamma)$, where t denotes

the parametrization variable, such that

$$D_t(fV) = \dot{f}V + fD_tV$$

for any smooth function f defined on $[0, 1]$, and

$$D_tV(t) = \nabla_{\dot{\gamma}(t)}\tilde{V}$$

for any extension $\tilde{V}$ of V on M . D_t is called the **covariant derivative operator** along γ , and in general it will play the role of “ $\frac{d}{dt}$ ” along curves in Riemannian manifold. We define the **acceleration** of γ to be the vector field $D_t\dot{\gamma}$ along γ , and γ is called a **geodesic** with respect to ∇ if its acceleration is zero, i.e.,

$$D_t\dot{\gamma} \equiv 0$$

on the interval $(0,1)$. This equation is called the **geodesic equation**. Such a definition of geodesics depends on the connection ∇ we use, so in order to have a concept of geodesics we must first choose a connection. It is a useful fact that every smooth manifold admits at least one linear connection. For example, embedded smooth submanifolds of Euclidean spaces have **tangential connection** $\nabla^\top$ [12, p.66]. For a Riemannian manifold M with metric g , however, there is a special one. A linear connection ∇ is said to be **compatible** with g if for all vector fields X, Y, Z

$$Xg(Y, Z) = g(\nabla_X Y, Z) + g(Y, \nabla_X Z).$$

Theorem 2. (Fundamental Lemma of Riemannian Geometry) *Let M be a Riemannian manifold with Riemannian metric g , there exists a unique linear connection ∇ on M that is compatible with g and symmetric.*

We refer to [12, Theorem 5.4] for the proof. This connection is called the **Riemannian connection** (a.k.a. **Levi-Civita connection**) of g . Geodesics respect to the Riemannian connection are called **Riemannian geodesics**. As we have anticipated, such geodesics has a close link to minimizing curves.

Theorem 3. *Every minimizing curve is a Riemannian geodesic when it is given unit speed parametrization.*

We again refer to [12, Theorem 6.6] for a proof that uses variational calculus. A simpler, albeit longer, proof can be found in [6]. The converse is almost true:

Theorem 4. *Every Riemannian geodesic is locally minimizing.*

See [12, Theorem 6.12] for the proof and a detailed discussion. Geodesics that happen to be minimizing is called **minimizing geodesics**. With this concept, we can see that the distance between two points is simply the length of the minimizing geodesic joining them. The distance function defined this way will be called the **Riemannian distance**. Since we will not use any other distance, the Riemannian distance will simply be called **distance**. Next, we shall review the distance formula on S^1 and S^{2n+1} , and then use them to construct the geodesic and the distance formula on $\mathbb{CP}^n$. The distance on S^1 and S^{2n+1} are denoted by d_{S^1} and $d_{S^{2n+1}}$ respectively, while our final goal, the distance on $\mathbb{CP}^n$, is simply denoted by d .

2.2.5.2 Distance on S^1 and S^{2n+1}

It is clear that the distance between two points $\mathbf{x} = (a, b)$ and $\mathbf{x}' = (1, 0)$ on the unit circle $S^1 \subset \mathbb{R}^2$ *has to be* the length of the shorter arch between the two points on the unit circle. This length is given by the angle between $\mathbf{x} = (a, b)$ and the horizontal axis on which $\mathbf{x}'$ lies:

$$d_{S^1}(\mathbf{x}, \mathbf{x}') = \cos^{-1}(a) = \cos^{-1} \begin{pmatrix} 1 \\ 0 \end{pmatrix}^\top \begin{pmatrix} a \\ b \end{pmatrix} = \cos^{-1}(\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{R}}) .$$

The same formula works in general for S^{2n+1} . For two distinct points on $\mathbf{x}, \mathbf{x}' \in S^{2n+1} \subset \mathbb{R}^{2n+2}$, there exists a unique 2-dimensional linear subspace of $\mathbb{R}^{2n+2}$ that contains both $\mathbf{x}$ and $\mathbf{x}'$. This subspace intersects S^{2n+1} on a circle. This circle is called the **great circle** through $\mathbf{x}$ and $\mathbf{x}'$. It is intuitively clear that the distance between them has to be the length of the shorter arc joining the two on the great circle (geodesic) that passes through both of them. Surprisingly, this seemingly obvious fact is not completely trivial to check. Here we refer to [12, Proposition 5.13] for an indirect proof that uses the fact that S^{2n+1} is homogeneous and isotropic. There exists an orthogonal change of coordinates after which the two points $\mathbf{x}$ and $\mathbf{x}'$ together with the great circle passing through them lie flat in $\mathbb{R}^2 \subset \mathbb{R}^{2n+2}$. Indeed, this change of coordinates is given explicitly by the QR decomposition:

there exists a $(2n + 2) \times (2n + 2)$ real orthogonal matrix Q such that

$$Q \begin{pmatrix} \mathbf{x} & \mathbf{x}' \end{pmatrix} = \begin{pmatrix} 1 & a \\ 0 & b \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix}$$

for some $a, b \in \mathbb{R}$. So the orthogonal transformation Q maps the great circle through those two points to the unit circle $S^1 \subset \mathbb{R}^2 \subset \mathbb{R}^{2n+2}$, and we can thus compute the distance as we did in the previous case. Since Q , being orthogonal, preserves dot product, we get

$$d_{S^{2n+1}}(\mathbf{x}, \mathbf{x}') = \cos^{-1}(a) = \cos^{-1}\left((Q\mathbf{x})^\top (Q\mathbf{x}')\right) = \cos^{-1}(\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{R}}) .$$

In this case, the distance is still given by the arccos of the real inner product of the two points as vectors in $\mathbb{R}^{2n+2}$.

2.2.5.3 Closest representatives of two points in $\mathbb{CP}^n$

For two points $[\mathbf{x}], [\mathbf{x}'] \in \mathbb{CP}^n \approx S^{2n+1}/U$, each of them has infinite number of representatives in S^{2n+1} of the form $e^{i\theta}\mathbf{x}$ and $e^{i\theta'}\mathbf{x}'$ respectively. We first need to find a pair with the minimum distance on S^{2n+1} . That is, we want to minimize the function $d_{S^{2n+1}}(e^{i\theta}\mathbf{x}, e^{i\theta'}\mathbf{x}')$

which is given by

$$\begin{aligned}
\cos^{-1}(\langle e^{i\theta} \mathbf{x}, e^{i\theta'} \mathbf{x}' \rangle_{\mathbb{R}}) &= \cos^{-1}(\operatorname{Re} \langle e^{i\theta} \mathbf{x}, e^{i\theta'} \mathbf{x}' \rangle_{\mathbb{C}}) \\
&= \cos^{-1}(\operatorname{Re} \overline{e^{i\theta}} \langle \mathbf{x}, e^{i\theta'} \mathbf{x}' \rangle_{\mathbb{C}}) \\
&= \cos^{-1}(\operatorname{Re} e^{-i\theta} \langle \mathbf{x}, e^{i\theta'} \mathbf{x}' \rangle_{\mathbb{C}}) \\
&= \cos^{-1}(\operatorname{Re} \langle \mathbf{x}, e^{i(\theta' - \theta)} \mathbf{x}' \rangle_{\mathbb{C}})
\end{aligned}$$

which, in turn, is simply $d_{S^{2n+1}}(\mathbf{x}, e^{i(\theta' - \theta)} \mathbf{x}')$. So it is enough to fix the representative $\mathbf{x}$ of $[\mathbf{x}]$ and minimize the distance over all the representatives of the other orbit $[\mathbf{x}']$. Hence, without loss of generality, we can focus on minimizing the real-valued function

$$d_{S^{2n+1}}(\mathbf{x}, e^{i\theta} \mathbf{x}') = \cos^{-1}(\langle \mathbf{x}, e^{i\theta} \mathbf{x}' \rangle_{\mathbb{R}})$$

over $\theta \in \mathbb{R}$. If we let r and ϕ be the real numbers such that $re^{i\phi} = \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}}$, i.e.,

$$r = |\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}}| \quad \text{and} \quad \phi = \arg \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}}$$

for some suitable branch of the complex argument function. Then $\langle \mathbf{x}, e^{i\theta} \mathbf{x}' \rangle_{\mathbb{R}}$ is simply

$$\operatorname{Re} \langle \mathbf{x}, e^{i\theta} \mathbf{x}' \rangle_{\mathbb{C}} = \operatorname{Re} \left(e^{i\theta} \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}} \right) = \operatorname{Re} \left(re^{i(\phi + \theta)} \right) = r \cos(\phi + \theta).$$

Therefore we want to minimize

$$d_{S^{2n+1}}(\mathbf{x}, e^{i\theta} \mathbf{x}') = \cos^{-1}(\langle \mathbf{x}, e^{i\theta} \mathbf{x}' \rangle_{\mathbb{R}}) = \cos^{-1}(r \cos(\phi + \theta))$$

over $\theta \in \mathbb{R}$. First of all, it is clear that if $r = 0$, then the above function is a constant function, and any θ would give us the “minimum value”. For $r = 1$, the above function is simply

$$\cos^{-1}(r \cos(\phi + \theta)) = \cos^{-1}(\cos(\phi + \theta)) = |\phi + \theta|$$

which is minimized at $\phi = -\theta$. Finally, by the Cauchy-Schwarz inequality $r = |\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}}| \leq \|\mathbf{x}\|_2 \cdot \|\mathbf{x}'\|_2 = 1$, so we only need to consider the case $0 < r < 1$. In this case, using basic calculus, to find the critical points we solve

$$\frac{d}{d\theta} \cos^{-1}(r \cos(\phi + \theta)) = \frac{r \sin(\phi + \theta)}{\sqrt{1 - (r \cos(\phi + \theta))^2}} = 0$$

which gives $\sin(\phi + \theta) = 0$. Hence $\theta \equiv -\phi \pmod{\pi}$. But since

$$\frac{d^2}{d\theta^2} \cos^{-1}(r \cos(\phi + \theta)) = \begin{cases} \frac{r}{\sqrt{1-r^2}} & \text{if } \theta \equiv -\phi \pmod{2\pi} \\ \frac{-r}{\sqrt{1-r^2}} & \text{if } \theta \equiv -\phi + \pi \pmod{2\pi} \end{cases}$$

the minimum is attained at $\theta \equiv -\phi \pmod{2\pi}$. We thus conclude that among all representatives $e^{i\theta} \mathbf{x}'$ of $[\mathbf{x}']$ in S^{2n+1}/U , $e^{-i\phi} \mathbf{x}'$ is the closest, in terms of the distance, given by $d_{S^{2n+1}}$, to $\mathbf{x}$. Furthermore, among all representatives of $[\mathbf{x}]$ and $[\mathbf{x}']$, $(\mathbf{x}, e^{-i\phi} \mathbf{x}')$ forms a pair with the minimum distance. For later reference, let us define

$$\eta_{\mathbf{x}}(\mathbf{x}') := e^{-i\phi} \mathbf{x}' \quad \text{where } \phi = \arg \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}},$$

with a suitable branch of the complex argument function, to be the nearest representative of $[\mathbf{x}']$ to $\mathbf{x}$ on S^{2n+1} . It is easy to see the important property

$$\langle \mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}') \rangle_{\mathbb{C}} = \langle \mathbf{x}, e^{-i\phi} \mathbf{x}' \rangle_{\mathbb{C}} = e^{-i\phi} \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}} = |\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}}| . \quad (2.5)$$

Based on this, we can easily derive

$$d_{S^{2n+1}}(\mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}')) = \cos^{-1}(\operatorname{Re} \langle \mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}') \rangle_{\mathbb{C}}) = \cos^{-1}(|\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}}|) . \quad (2.6)$$

Next, we shall show that this distance also gives us the distance between $[\mathbf{x}]$ and $[\mathbf{x}']$ in $\mathbb{CP}^n = S^{2n+1}/U$.

2.2.5.4 Geodesics in $\mathbb{CP}^n$

We start with finding out the geodesic between $[\mathbf{x}]$ and $[\mathbf{x}']$ in $\mathbb{CP}^n$. We have just shown that among infinitely many representatives in S^{2n+1} the two points $\mathbf{x}$ and $\eta_{\mathbf{x}}(\mathbf{x}')$ form a closest pair in terms of their distance in S^{2n+1} . Let $\gamma : [0, 1] \rightarrow S^{2n+1}$ be the minimizing Riemannian geodesic (the shorter arc of a great circle) in S^{2n+1} joining $\mathbf{x}$ and $\eta_{\mathbf{x}}(\mathbf{x}')$ with $\gamma(0) = \mathbf{x}$, $\gamma(1) = \eta_{\mathbf{x}}(\mathbf{x}')$. We wish to show the smooth function $\hat{\gamma}(t) := [\gamma(t)]$ parametrizes the minimizing Riemannian geodesic in $\mathbb{CP}^n$ joining $[\mathbf{x}]$ and $[\mathbf{x}']$, and the two curves have the same length. This can be accomplished in three steps. First, we shall prove that the tangent vector of γ is always in the horizontal subspace $H_{\gamma(t)}$ of $T_{\gamma(t)}S^{2n+1}$. In this case, γ is said to be **horizontal**. With this, we will then show that $\hat{\gamma}$ is indeed a Riemannian geodesic in $\mathbb{CP}^n$. Finally, we need to establish the minimizing property of the geodesic $\hat{\gamma}$.

By construction, at each point $\gamma(t) \in S^{2n+1}$ of the curve, the tangent space can be

decomposed into orthogonal subspaces

$$T_{\gamma(t)}S^{2n+1} = V_{\gamma(t)} \oplus H_{\gamma(t)} \quad \text{where} \quad V_{\gamma(t)} = T_{\gamma(t)}[\gamma(t)] = \text{span}\{i\dot{\gamma}(t)\}.$$

In particular, at $t = 1$, we have $V_{\gamma(1)} = \text{span}\{i\dot{\gamma}(1)\} = \text{span}\{i\eta_{\mathbf{x}}(\mathbf{x}')\}$ which give us the direction of the orbit $[\mathbf{x}']$. Also by definition, the great circle passing through $\mathbf{x}$ and $\eta_{\mathbf{x}}(\mathbf{x}')$, as a set, is the intersection between S^{2n+1} and the unique 2-dimensional linear subspace $L = \text{span}\{\mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}')\}$ of $\mathbb{R}^{2n+2}$ that contains both $\mathbf{x}$ and $\eta_{\mathbf{x}}(\mathbf{x}')$. Then γ , being a segment of this great circle, must also lie flat inside the subspace L , and hence L must contain the tangent vector $\dot{\gamma}(1)$. So to prove $\dot{\gamma}(1)$ is orthogonal to $V_{\gamma(1)}$, it is enough to show that the vertical space $V_{\gamma(1)}$ is orthogonal to the basis $\{\mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}')\}$ of L which contains $\dot{\gamma}(1)$. Notice that

$$\begin{aligned} \langle \mathbf{x}, i\eta_{\mathbf{x}}(\mathbf{x}') \rangle_{\mathbb{R}} &= \text{Re} \langle \mathbf{x}, i\eta_{\mathbf{x}}(\mathbf{x}') \rangle_{\mathbb{C}} \\ &= \text{Re}(i \langle \mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}') \rangle_{\mathbb{C}}) \\ &= \text{Re}(i |\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}}|) \\ &= 0 \end{aligned}$$

by Equation (2.5), and similarly,

$$\begin{aligned} \langle \eta_{\mathbf{x}}(\mathbf{x}'), i\eta_{\mathbf{x}}(\mathbf{x}') \rangle_{\mathbb{R}} &= \text{Re} \langle \eta_{\mathbf{x}}(\mathbf{x}'), i\eta_{\mathbf{x}}(\mathbf{x}') \rangle_{\mathbb{C}} \\ &= \text{Re}(i \langle \eta_{\mathbf{x}}(\mathbf{x}'), \eta_{\mathbf{x}}(\mathbf{x}') \rangle_{\mathbb{C}}) \\ &= \text{Re}(i) \\ &= 0. \end{aligned}$$

Therefore the generator $i\dot{\gamma}(1) = i\eta_{\mathbf{x}}(\mathbf{x}')$ of $V_{\gamma(1)}$ is orthogonal to both $\mathbf{x}$ and $\eta_{\mathbf{x}}(\mathbf{x}')$ which are the basis of the subspace L . This implies $V_{\gamma(1)}$ is orthogonal to the entire subspace $L = \text{span}\{\mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}')\}$. In particular, it is orthogonal to $\dot{\gamma}(1) \in L$. Therefore $\dot{\gamma}(1)$ lies in the horizontal subspace $H_{\gamma(1)}$. Repeating the same argument for any other $t \in [0, 1]$, so $\dot{\gamma}(t) \in H_{\gamma(t)}$ for all $t \in [0, 1]$. With this observation, we can now apply the following theorem [6, Proposition 2.109]:

Theorem 5.

- i. If γ is a Riemannian geodesic of S^{2n+1} and $\dot{\gamma}(1)$ is horizontal, then the curve $\hat{\gamma} = [\gamma(t)]$ is a Riemannian geodesic of S^{2n+1}/U of the same length as γ .*
- ii. Conversely, if $\hat{\gamma}$ is a Riemannian geodesic of S^{2n}/U with $\hat{\gamma}(0) = [\mathbf{x}]$, then there exists a unique Riemannian geodesic γ in S^{2n+1} such that $\gamma(0) = \mathbf{x}$, $[\gamma(t)] = \hat{\gamma}(t)$, and it is horizontal in the sense that $\dot{\gamma}(t) \in H_{\gamma(t)}$. In this case, γ is called the **horizontal lift** of $\hat{\gamma}$.*

Hence $\hat{\gamma}$ is a Riemannian geodesic of $\mathbb{CP}^n = S^{2n+1}/U$ joining the two points $[\mathbf{x}]$ and $[\mathbf{x}']$. To show this geodesic is indeed minimizing, suppose there is a shorter minimizing curve $\hat{\gamma}_2 : [0, 1] \rightarrow \mathbb{CP}^n$ joining $[\mathbf{x}]$ and $[\mathbf{x}']$. Without loss of generality, let us assume it has constant speed parametrization. By Theorem 3, $\hat{\gamma}_2$ is a Riemannian geodesic. Then the second part of Theorem 5 implies that there is a unique Riemannian geodesic $\gamma_2 : [0, 1] \rightarrow S^{2n+1}$, the horizontal lift of $\hat{\gamma}_2$, such that $\gamma_2(0) = \mathbf{x}$, $\hat{\gamma}_2 \equiv [\gamma_2]$, and $\dot{\gamma}_2(t) \in H_{\gamma_2(t)}$ for $t \in [0, 1]$. Let $\mathbf{x}'' := \gamma_2(1)$, then $[\mathbf{x}''] = [\mathbf{x}']$ by construction. Let us still use $\ell_{\mathbb{CP}^n}$ and $\ell_{S^{2n+1}}$ as the operators that assign length to smooth curves. Since γ_2 is horizontal, we necessarily have $\ell_{S^{2n+1}}(\gamma_2) = \ell_{\mathbb{CP}^n}(\hat{\gamma}_2)$ because the quotient map $\pi : S^{2n+1} \rightarrow S^{2n+1}/U \approx \mathbb{CP}^n$ is a

Riemannian submersion, thus

$$\begin{aligned}
d_{S^{2n+1}}(\mathbf{x}, \mathbf{x}'') &= \ell_{S^{2n+1}}(\gamma_2) \\
&= \ell_{\mathbb{CP}^n}(\hat{\gamma}_2) \\
&< \ell_{\mathbb{CP}^n}(\hat{\gamma}) \\
&= \ell_{S^{2n+1}}(\gamma) \\
&= d_{S^{2n+1}}(\mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}')) .
\end{aligned}$$

This provides another representative $\mathbf{x}''$ of $[\mathbf{x}']$ that is closer to $\mathbf{x}$ than $\eta_{\mathbf{x}}(\mathbf{x}')$, a contradiction. We conclude that such a shorter minimizing curve joining $[\mathbf{x}]$ and $[\mathbf{x}']$ cannot exist, and thus $\hat{\gamma}$, as constructed, is the true minimizing Riemannian geodesic joining $[\mathbf{x}]$ and $[\mathbf{x}']$ with which the distance between them can be defined.

2.2.5.5 Summary: the distance formula in $\mathbb{CP}^n$

The distance $d([\mathbf{x}], [\mathbf{x}'])$ between the points $[\mathbf{x}], [\mathbf{x}'] \in \mathbb{CP}^n$ is simply the length of the minimizing Riemannian geodesic $\hat{\gamma}$ joining the two which, as we have just derived, is the same as the length of γ :

$$d([\mathbf{x}], [\mathbf{x}']) = \ell_{\mathbb{CP}^n}(\hat{\gamma}) = \ell_{S^{2n+1}}(\gamma) = d_{S^{2n+1}}(\mathbf{x}, \eta_{\mathbf{x}}(\mathbf{x}')) .$$

Therefore we have the distance formula

$$d([\mathbf{x}], [\mathbf{x}']) = \cos^{-1}(|\langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}}|) . \quad (2.7)$$

2.3 Projective predictors

For some fixed $t_0 \in [0, 1]$, given a representative $(\mathbf{x}, t_0)$ of a point with $[\mathbf{x}] \in \mathbb{CP}^n$ on a projective homotopy path, i.e., $\hat{H}(\mathbf{x}, t_0) = 0$, the job of a predictor is to produce a prediction $\mathbf{x}'$ so that the point $(\mathbf{x}', t_0 + \Delta t)$, for some given step size $\Delta t > 0$, represents a point that is *close to* being on that projective path. The simplest possible predictor is the zero predictor, which actually does nothing. More precisely, the zero predictor produces $(\mathbf{x}, t_0 + \Delta t)$ as the prediction from $(\mathbf{x}, t_0)$. [4] has demonstrated the effectiveness of such a predictor. However such method is in general too inefficient to be used in practice.

2.3.1 Projective Euler's method

The projective Euler's predictor produces the prediction $\mathbf{x}'$ by moving $\mathbf{x}$ one step toward the tangent direction of $[\mathbf{x}(t)]$, viewed as a smooth curve in $\mathbb{CP}^n$. This requires three steps. First let $\mathbf{v}$ be the solution of the system of equation

$$\begin{aligned} \left[\hat{H}_{\mathbf{x}}(\mathbf{x}, t) \right] \mathbf{v} &= -\hat{H}_t(\mathbf{x}, t) \\ \langle \sigma \mathbf{x}, \mathbf{v} \rangle_{\mathbb{C}} &= 0 \end{aligned}$$

where $\sigma \in [\sigma_n, \sigma_1]$, called the **conditioning factor**, is an arbitrary real number chosen between σ_n and σ_1 , the n -th and the first singular values of the matrix $\hat{H}_{\mathbf{x}}(\mathbf{x}, t)$. The solvability of the above linear system and the role σ plays here will be discussed in detail later. We shall simply state here that under the smoothness assumption of the homotopy this system is numerically solvable. Next, we let

$$\tilde{\mathbf{x}} = \mathbf{x} + \mathbf{v} \cdot \Delta t$$

where Δt is the step size. Finally we take

$$\mathbf{x}' = \tilde{\mathbf{x}} / \|\tilde{\mathbf{x}}\|_2$$

to be the prediction, which is obvious inside S^{2n+1} . To express this method more formally, let us define the map $\mathcal{R} : \mathbb{C}^{n+1} \rightarrow S^{2n+1}$ given by

$$\mathcal{R}(\mathbf{x}) = \frac{\mathbf{x}}{\|\mathbf{x}\|_2}.$$

Then $\mathcal{R}$ is a retraction of $\mathbb{C}^{n+1}$, taking away the origin, to the subset S^{2n+1} , and the prediction $\mathcal{E}(\mathbf{x}, \Delta t) := \mathbf{x}'$ of the projective Euler's method can be expressed as

$$\mathcal{E}(\mathbf{x}, \Delta t) = \mathcal{R} \left(\mathbf{x} - \Delta t \begin{pmatrix} \hat{H}_{\mathbf{x}}(\mathbf{x}, t) \\ \sigma \mathbf{x}^H \end{pmatrix}^{-1} \begin{pmatrix} \hat{H}_t(\mathbf{x}, t) \\ 0 \end{pmatrix} \right).$$

2.4 Projective correctors

The prediction $(\mathbf{x}', t_0 + \Delta t)$ produced by a predictor may not be exactly on or even very close to the path defined by $\hat{H} = \mathbf{0}$. If the next prediction step is to start from such a poor approximation, the error can quickly build up to an unacceptable level. To curb such error accumulation, a corrector is needed to return the prediction to the path. If we now fix $t_1 = t_0 + \Delta t$, the equation $\hat{H} = \mathbf{0}$ becomes a system of n homogeneous polynomial equations in $n + 1$ unknowns. The job of a corrector is to produce a refinement $\mathbf{x}''$ of the approximate solution $\mathbf{x}'$ of $\hat{H} = \mathbf{0}$ at $t = t_1$. For correctors, failure is always an option. When a corrector fails to bring the prediction back to the path quickly and reliably, it is usually the case

that the step size Δt used in the prediction step is too large, and the prediction should be performed again with a smaller step size.

A natural choice of the corrector is an extension of the Newton's iteration to the complex projective space. In the following subsections, we will outline this method, developed in [24], and two related methods.

2.4.1 Projective Newton's method

In [24] MICHAEL SHUB and STEVEN SMALE proposed the Projective Newton's method given by

$$\mathcal{N}(\mathbf{x}) = \mathcal{R} \left(\mathbf{x} - \begin{pmatrix} \hat{H}_{\mathbf{x}}(\mathbf{x}, t) \\ \sigma \mathbf{x}^H \end{pmatrix}^{-1} \begin{pmatrix} \hat{H}(\mathbf{x}, t) \\ 0 \end{pmatrix} \right).$$

Notice that a conditioning factor σ , identical to the one used in the projective Euler's method, is introduced to improve the numerical stability of the method. The role it plays here will be discussed in Section 2.6. Using the starting point $[\mathbf{x}^0] = [\mathbf{x}']$ produced by a predictor, we can perform projective Newton's iterations

$$\mathbf{x}^k = \mathcal{N}(\mathbf{x}^{k-1})$$

for $k = 1, 2, \dots$ until certain stopping criteria are met. One can interpret a single step of projective Newton's iteration geometrically as a single step of the regular Newton's iteration restricted to the hyperplane of $\mathbb{C}^{n+1}$ defined by the tangent space of $\mathbb{CP}^n$ at that point.

2.4.1.1 Convergence property

It is more convenient to discuss the convergence property if we use another measure of the “distance” as defined in [24]

$$d_T([\mathbf{x}], [\mathbf{x}']) := \tan d([\mathbf{x}], [\mathbf{x}']) = \|\mathbf{x} - \eta_{\mathbf{x}}(\mathbf{x}')\|_2 .$$

With this, it has been shown in [4] that for an exact regular isolated solution $[\zeta]$ of $F([\zeta]) = \hat{H}(\zeta, t) = \mathbf{0}$ there is a neighborhood U containing $[\zeta]$ of $\mathbb{CP}^n$ such that if $[\mathbf{x}^0] \in U$ then

$$d_T([\zeta], [\mathbf{x}^k]) \leq \left(\frac{1}{2}\right)^{2^k - 1} d_T([\zeta], [\mathbf{x}^0])$$

for $k = 1, 2, \dots$. So with d_T used as the “distance”, the projective Newton’s method can be considered having quadratic convergence property when the above condition is satisfied. We refer to [4] for the proof as well as the exact statement. When the actual Riemannian distance $d = d_{\mathbb{CP}^n}$ is used, the above formula becomes much more complicated. Nevertheless, we can still conclude, qualitatively, that as long as the starting point $[\mathbf{x}^0]$ is sufficiently close to the solution $[\zeta]$, the projective Newton’s method converges very quickly.

2.4.1.2 Stopping criteria

The Newton’s method, being an iterative method, can be repeated indefinitely. Stopping criteria are used to mark the termination of the algorithm. Following the point of view that a projective Newton’s iteration is simply a regular Newton’s iteration restricted to certain hyperplanes, we expect the existing stopping criteria developed for the Newton’s corrector in the context of affine path tracking to be usable with minimal modifications:

- First of all, a small residual is an obvious stopping criteria. The residual, a numerical measure of the closeness of a point from being a solution, is the topic of Section 2.7. If the residual at $[\mathbf{x}^k]$ is sufficiently small, the correction is considered to have succeeded. This threshold depends on both the path condition and the machine epsilon.
- If the Riemannian distance $d([\mathbf{x}^{k+1}], [\mathbf{x}^k])$ between consecutive points $[\mathbf{x}^{k+1}]$ and $[\mathbf{x}^k]$ is greater than a certain threshold, then the correction is considered to have failed. This criterion is used to prevent a phenomenon commonly known as “curve jumping” in which the path tracking algorithm accidentally switch paths. We refer to [15] for the discussion of this rare but potentially dangerous phenomenon. The threshold used here is proportional to the quotient between the volume of S^{2n+1} and the number of paths defined by the homotopy.
- Finally, if the sequence of points $[\mathbf{x}^1], [\mathbf{x}^2], \dots$ in $\mathbb{CP}^n$ does not stabilize, in the sense of Riemannian distance d , within a few iterations, we consider the correction to have failed. In our actual implementation the limit on the number of iterations range from 3 to 7 depending on the path condition.

We refer to [15] for the complete list of the stopping criteria as well as their detailed descriptions. Our preliminary implementation, equipped with these stopping criteria, has shown competitive performance, as we shall see in the last section of this chapter.

2.4.2 Projective Newton’s method with damping factor

The regular Newton’s method is often modified with a damping factor to improve its convergence property. The same modification can be applied to the projective Newton’s

method given a damping factor $\lambda \in [0, 1]$, given by

$$\mathcal{N}_\lambda(\mathbf{x}) = \mathcal{R} \left(\mathbf{x} - \lambda \begin{pmatrix} \hat{H}_{\mathbf{x}}(\mathbf{x}, t) \\ \sigma \mathbf{x}^H \end{pmatrix}^{-1} \begin{pmatrix} \hat{H}(\mathbf{x}, t) \\ 0 \end{pmatrix} \right).$$

Clearly, the projective Newton's method can be considered as the special case with $\lambda = 1$, i.e., $\mathcal{N} = \mathcal{N}_1$. Just like the Newton's method, the dampened version is repeated with the starting point $[\mathbf{x}^0] = [\mathbf{x}']$ produced by a predictor:

$$\mathbf{x}^k = \mathcal{N}_\lambda(\mathbf{x}^{k-1})$$

for $k = 1, 2, \dots$ potentially with non-constant damping factor λ , until certain stopping criteria are met. Our preliminary implementation adopts the same stopping criteria used for the projective Newton's corrector, and the damping factor ranges from 0.5 to 1.0 depending on the number of consecutive failed corrections. We refer to [26] for the general theory and convergence analysis.

2.5 Projective Newton's homotopy

Another corrector is the projective Newton's homotopy method. In this method one solves the correction problem via another homotopy continuation method. Since in the correction problem we fix $t_1 = t_0 + \Delta t$, we can define $\hat{F}(\mathbf{x}) = \hat{H}(\mathbf{x}, t_1)$ which is a nonlinear system of n equations in $n + 1$ unknowns. The prediction $\mathbf{x}'$ is assumed or at least expected to be close to a true solution $[\zeta]$ of $\hat{F}(\mathbf{x}) = \mathbf{0}$ in the complex projective space which the corrector intends to locate. In order to do so, we can consider the homotopy $\hat{\mathcal{H}}^{t_1} : \mathbb{C}^{n+1} \times [0, 1] \rightarrow \mathbb{C}^{n+1}$ given

by

$$\hat{\mathcal{H}}^{t_1}(\mathbf{x}, s) = \begin{cases} \hat{F}(\mathbf{x}) - (1-s)\hat{F}(\mathbf{x}') & = \hat{H}(\mathbf{x}, t_1) - (1-s)\hat{H}(\mathbf{x}', t_1) \\ \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}} - 1 & = \langle \mathbf{x}, \mathbf{x}' \rangle_{\mathbb{C}} - 1 \end{cases}.$$

Clearly, at $s = 0$, $\hat{\mathcal{H}}^{t_1}(\mathbf{x}', 0) = \mathbf{0}$, and at $s = 1$, the solution set of $\hat{\mathcal{H}}^{t_1}(\mathbf{x}, 1) = \mathbf{0}$ may contain some representatives of $[\zeta]$. Standard path tracking techniques can be used to tracking the solution path from $s = 0$ at $\mathbf{x}'$ to $s = 1$.

Just like the Newton's iteration, the projective Newton's homotopy may or may not converge. The convergence property and the cost analysis can be found in [1]. While generally much more expensive, as an experimentally verified rule of thumb, the projective Newton homotopy generally offers much larger region of fast convergence than the projective Newton's method. Hence we suspect that this method has its own niche among the correctors.

2.6 Numerical analysis of the algorithms

Theoretically, for the above algorithms to work, the square linear system of equations in $\mathbf{v} \in \mathbb{C}^{n+1}$

$$\begin{aligned} \left[\hat{H}_{\mathbf{x}}(\mathbf{x}, t) \right] \cdot \mathbf{v} &= \mathbf{b} \\ \langle \sigma \mathbf{x}, \mathbf{v} \rangle_{\mathbb{C}} &= 0 \end{aligned}$$

must be solvable for any vector $\mathbf{b} \in \mathbb{C}^n$ and a pair $(\mathbf{x}, t)$ representing a point $([\mathbf{x}], t)$ on a projective path defined by $\hat{H} = \mathbf{0}$. Moreover, since these are intended to be numerical algorithms, we must also show the above system is well conditioned. For this fixed t , define $F(\mathbf{x}) = \hat{H}(\mathbf{x}, t) = (f_1, \dots, f_n)$, then the above equation can be written as

$$\begin{pmatrix} DF(\mathbf{x}) \\ \sigma \mathbf{x}^H \end{pmatrix} \mathbf{v} = \begin{pmatrix} \mathbf{b} \\ 0 \end{pmatrix}$$

where $DF(\mathbf{x})$ is the Jacobian matrix of F respect to $\mathbf{x}$. Since $F(\mathbf{x}) = (f_1, \dots, f_n)$ is a system of homogeneous polynomials and $F(\mathbf{x}) = \mathbf{0}$, by the Euler's theorem (1) for homogeneous functions,

$$\sum_{j=0}^n x_j \frac{\partial f_i}{\partial x_j} = d_i \cdot f_i(\mathbf{x}) = 0$$

for each $i = 1, \dots, n$ where $d_i = \deg f_i$. Letting $A = DF(\mathbf{x})$, we have $A\mathbf{x} = \mathbf{0}$. In other words, $\mathbf{x} \in \ker A$. Then we can find n right singular vectors $\{\mathbf{v}^1, \dots, \mathbf{v}^n\}$ such that $\{\mathbf{v}^1, \dots, \mathbf{v}^n, \mathbf{x}\}$ form an orthonormal basis of $\mathbb{C}^{n+1}$ with respect to the complex inner product and A has the

singular value decomposition of the form

$$U^H A \begin{pmatrix} \mathbf{v}^1 & \dots & \mathbf{v}^n & \mathbf{x} \end{pmatrix} = \begin{pmatrix} \sigma_1 & & & 0 \\ & \ddots & & \vdots \\ & & \sigma_n & 0 \end{pmatrix}$$

for some unitary $n \times n$ matrix U , where $\{\sigma_1, \dots, \sigma_n\}$ with $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$ are the first n singular values of A . It follows that

$$\begin{aligned} \begin{pmatrix} U^H \\ 1 \end{pmatrix} \begin{pmatrix} A \\ \sigma \mathbf{x}^H \end{pmatrix} \begin{pmatrix} \mathbf{v}^1 & \dots & \mathbf{v}^n & \mathbf{x} \end{pmatrix} &= \begin{pmatrix} U^H A \begin{pmatrix} \mathbf{v}^1 & \dots & \mathbf{v}^n & \mathbf{x} \end{pmatrix} \\ \sigma \mathbf{x}^H \begin{pmatrix} \mathbf{v}^1 & \dots & \mathbf{v}^n & \mathbf{x} \end{pmatrix} \end{pmatrix} \\ &= \begin{pmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_n & \\ & & & \sigma \end{pmatrix}. \end{aligned}$$

Simply put, the matrix $\begin{pmatrix} DF(\mathbf{x}) \\ \sigma \mathbf{x}^H \end{pmatrix} = \begin{pmatrix} A \\ \sigma \mathbf{x}^H \end{pmatrix}$ has singular values $\sigma_1, \dots, \sigma_n$, and σ which is within the interval $[\sigma_n, \sigma_1]$ by our construction. Thus the maximum and the minimum singular value of $\begin{pmatrix} DF(\mathbf{x}) \\ \sigma \mathbf{x}^H \end{pmatrix}$ are still σ_1 and σ_n respectively, and its condition number is

$$\text{cond} \begin{pmatrix} DF(\mathbf{x}) \\ \sigma \mathbf{x}^H \end{pmatrix} = \frac{\sigma_1}{\sigma_n}.$$

Theoretically, at least, by the smoothness condition of the homotopy construction,

$DF(\mathbf{x}) = \hat{H}_{\mathbf{x}}(\mathbf{x}, t)$ must be of full row rank and thus $\sigma_n \neq 0$. This implies that the matrix $\begin{pmatrix} DF(\mathbf{x}) \\ \sigma \mathbf{x}^H \end{pmatrix}$ is nonsingular. But more importantly, its condition number is exactly $\frac{\sigma_1}{\sigma_n}$ which is determined by the original homotopy construction. Since the path condition at $(\mathbf{x}, t)$ is defined to be the condition number of this matrix, we can conclude that our projective path tracking algorithms *do not pollute the path condition*.

2.7 Numerical determination of residual

In the affine path tracking process, one tracks a smooth path defined by $H(\mathbf{x}, t) = \mathbf{0}$ where $H : \mathbb{C}^n \times [0, 1] \rightarrow \mathbb{C}^n$. However, it is usually not possible to stay exactly on that path due to truncation error from the numerical algorithms used or rounding error caused by floating point computation. The best one could hope is, during the path tracking process, we have $H(\mathbf{x}, t) \approx \mathbf{0}$ at each step. In this context, the magnitude of the norm $\|H(\mathbf{x}, t)\|$ is generally a good measure of how far the point $(\mathbf{x}, t) \in \mathbb{C}^n \times [0, 1]$ is from the path in the sense that when $\|H(\mathbf{x}, t)\| = 0$ it is definitely on the path, and when $\|H(\mathbf{x}, t)\|$ is very large then it is very unlikely to be close to the path. Hence this norm $\|H(\mathbf{x}, t)\|$, called **residual**, is usually a good numerical indicator of the closeness of a point to paths defined by $H(\mathbf{x}, t) = \mathbf{0}$. Throughout the path tracking process, one must make sure the residual is small.

In this section, the notion of the residual for projective path tracking will be developed. Just like in the affine case, for a fixed t , the condition $\hat{H}(\mathbf{x}, t) = \mathbf{0} \in \mathbb{C}^n$ is a clear indication that $[\mathbf{x}] \in \mathbb{CP}^n$ is indeed a solution, in $\mathbb{CP}^n$, to the homogeneous system $\hat{H}(\mathbf{x}, t) = \mathbf{0}$. Unfortunately, $\hat{H}(\mathbf{x}, t) \approx \mathbf{0}$ does not imply $(\mathbf{x}, t)$ is close to being a solution to $\hat{H}(\mathbf{x}, t) = \mathbf{0}$. To see this, note that the norm $\|\hat{H}(\mathbf{x}, t)\|$ is not generally meaningful: the values of homogeneous polynomials are not well defined since for a homogeneous function $f : \mathbb{C}^{n+1} \rightarrow \mathbb{C}$ of

degree d , $f(\lambda \mathbf{x}) = \lambda^d f(\mathbf{x})$, while $\mathbf{x}$ and $\lambda \mathbf{x}$ in $\mathbb{C}^{n+1}$ actually represent the same point in $\mathbb{CP}^n = (\mathbb{C}^{n+1} \setminus \{\mathbf{0}\}) / \sim$.

2.7.1 Affine residual

In many cases, the original residual $\|H(\mathbf{x}, t)\|$ is what the users really care about. By definition $H(x_1, \dots, x_n, t) = \hat{H}(1, x_1, \dots, x_n, t)$. For a point $[\mathbf{x}]$ in $\mathbb{CP}^n = (\mathbb{C}^n \setminus \{\mathbf{0}\}) / \sim$ with homogeneous coordinate $\mathbf{x} = [x_0 : \dots : x_n]$ with $x_0 \neq 0$, it is equivalent to $\left[1 : \frac{x_1}{x_0} : \dots : \frac{x_n}{x_0}\right]$, so the **affine residual** is

$$\begin{aligned} \rho_A(\mathbf{x}) &:= \left\| H\left(\frac{x_1}{x_0}, \dots, \frac{x_n}{x_0}, t\right) \right\|_2 \\ &= \left\| \hat{H}\left(\frac{x_0}{x_0}, \frac{x_1}{x_0}, \dots, \frac{x_n}{x_0}, t\right) \right\|_2 \\ &= \left\| \begin{pmatrix} \frac{1}{d_1} \hat{h}_1(x_0, \dots, x_n, t) \\ \vdots \\ \frac{1}{d_n} \hat{h}_n(x_0, \dots, x_n, t) \end{pmatrix} \right\|_2 \end{aligned}$$

which is equivalent to a weighted norm $\|\bullet\|_{W(\mathbf{x})}$ with weights $W(\mathbf{x}) = (1/|x_0|^{d_1}, \dots, 1/|x_n|^{d_n})$ given by

$$\left\| \hat{H}(\mathbf{x}, t) \right\|_{W(\mathbf{x})} := \sqrt{\sum_{k=1}^n \left(\left| \hat{h}_k(\mathbf{x}, t) \right| / \left| x_0^{d_k} \right| \right)^2}.$$

Thus we can use $\|\hat{H}(\mathbf{x}, t)\|_{W(\mathbf{x})}$ as a measure of how close $\mathbf{x}$ is from being a solution of $\hat{H} = \mathbf{0}$ at the fixed t -value whenever x_0 is not close to zero. Similarly, the relative residual is

given by

$$\frac{\|\hat{H}(\mathbf{x}, t)\|_{W(\mathbf{x})}}{\left\|\begin{pmatrix} x_1 \\ x_0 \end{pmatrix}, \dots, \begin{pmatrix} x_n \\ x_0 \end{pmatrix}\right\|} = \frac{\sqrt{\sum_{k=1}^n \left(|\hat{h}_k(\mathbf{x}, t)| / |x_0^{d_k}| \right)^2}}{\|(x_1, \dots, x_n)\| / |x_0|} = \frac{\sqrt{\sum_{k=1}^n \left(|\hat{h}_k(\mathbf{x}, t)| / |x_0^{d_k-1}| \right)^2}}{\|(x_1, \dots, x_n)\|}.$$

The obvious downside of this affine residual is that it is undefined for any point with $x_0 = 0$.

For any point with x_0 close to being zero, the computation of the affine residual as defined above is numerically unstable.

2.7.2 Tangential residual

In many cases it is useful to restrict ourselves to the projection of $\mathbb{CP}^n = (\mathbb{C}^{n+1} \setminus \{\mathbf{0}\}) / \sim$ on a hyperplane of $\mathbb{C}^{n+1}$ as we did when using (2.2).

For a fixed t and $[\mathbf{x}^0] \in \mathbb{CP}^n$ that is known to be close to the actual solution of $\hat{H}(\mathbf{x}, t) = \mathbf{0}$ one is seeking, the hyperplane determined by the tangent space at this point $\mathbf{x}^0 + T_{[\mathbf{x}^0]} \mathbb{CP}^n$ can be used as a hyperplane to which we project $\mathbb{CP}^n = (\mathbb{C}^{n+1} \setminus \{\mathbf{0}\}) / \sim$, and define the **tangential residual**

$$\rho_T(\mathbf{x}) := \left\| \begin{pmatrix} \hat{H}(\mathbf{x}, t) \\ \left\langle \frac{\mathbf{x}^0}{\|\mathbf{x}^0\|_2}, \mathbf{x} \right\rangle_{\mathbb{C}} - 1 \end{pmatrix} \right\|_2$$

which implicitly depends on the choices of the point $\mathbf{x}^0$. Unlike the affine residual ρ_A , the tangential residual is defined for any $[\mathbf{x}]$ sufficiently close to $[\mathbf{x}^0]$ in terms of the Riemannian distance. An apparent downside is that different representatives $\lambda \mathbf{x}$ of the point $[\mathbf{x}]$ will produce different residual. In our experiments, however, we found this almost never causes problems.

2.7.3 Spherical residual

Another residual stays true to our point of view of the complex projective space $\mathbb{CP}^n \approx S^{2n+1}/U$ is what we shall call the **spherical residual** ρ_S . While $\|\hat{H}(\mathbf{x}, t)\|_2$ is not well defined for $[\mathbf{x}] \in \mathbb{CP}^n = (\mathbb{C}^{n+1} \setminus \{\mathbf{0}\})/\sim$, the value

$$\rho_S(\mathbf{x}) = \left\| \hat{H}(\mathbf{x}, t) \right\|_2$$

is, however, well defined on S^{2n+1}/U . This is because $[\mathbf{x}] := \{e^{i\theta}\mathbf{x} | \theta \in \mathbb{R}\}$ in S^{2n+1}/U , and ρ_S is clearly invariant under such action.

2.7.4 Summary of residuals

Each of the definitions for residuals has different properties and its own pros and cons:

- The affine residual ρ_A is usually what the users actually care about, however it is undefined for any points with $x_0 = 0$.
- The tangential residual ρ_T is always computable, but it is, in general, not well defined.
- The spherical residual ρ_S is always well defined on S^{2n+1}/U , but it has no connection to the affine residual the users care about.

In our preliminary implementation, a combination of these residuals is used. Inside Newton-like correctors the tangential residual ρ_T is used with the role of $\mathbf{x}^0$ played by the most recent $\mathbf{x}$ -value produced by the iteration. In all other places during the path tracking process, the spherical residual ρ_S is used. In the final refinement stage, in which the solutions are “refined” to a higher accuracy, if we are confident that a solution $[x_0 : \dots : x_n]$ is in the affine part of $\mathbb{CP}^n$, i.e., $x_0 \neq 0$, then ρ_A is preferred.

2.8 Special conditioning

2.8.1 The curious case of **barry**

Figure 2.4 shows the path condition along a single path when our projective path tracking algorithm on S^{2n+1}/U is used to solve the **barry** problem from the POSSo [2] test suite. With only 3 equations and 3 unknowns, **barry** is one of the smallest systems in the POSSo test suite. We thus expect it being very easy to solve.

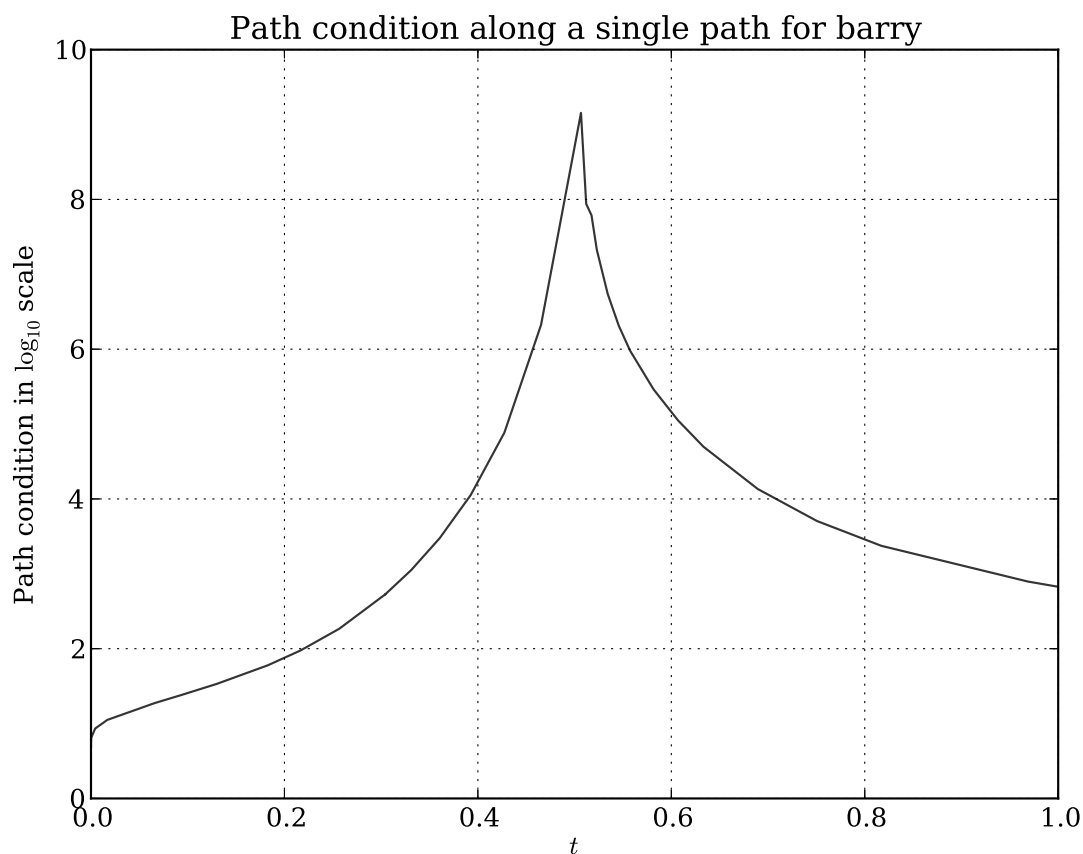


Figure 2.4: Path condition, in $\log_{10}$ scale, along a single path tracked for solving the **barry** problem using projective path tracking algorithms on S^{2n+1}/U . The maximum path condition exceeds 10^9 .

The maximum path condition exceeds 10^9 . While our path tracking algorithm with

double-precision floating point arithmetic had no trouble tracking this path, considering the simplicity of the system, this result is certainly surprising and, to a certain extent, unsettling. It turns out, the bad numerical path condition is caused by unusually large difference in the scales of the rows of the Jacobian matrix. Figure 2.5 shows the large difference in the scales of the rows in the Jacobian matrix along a single path when the projective path tracking is used. Clearly, this would naturally cause terrible path condition.

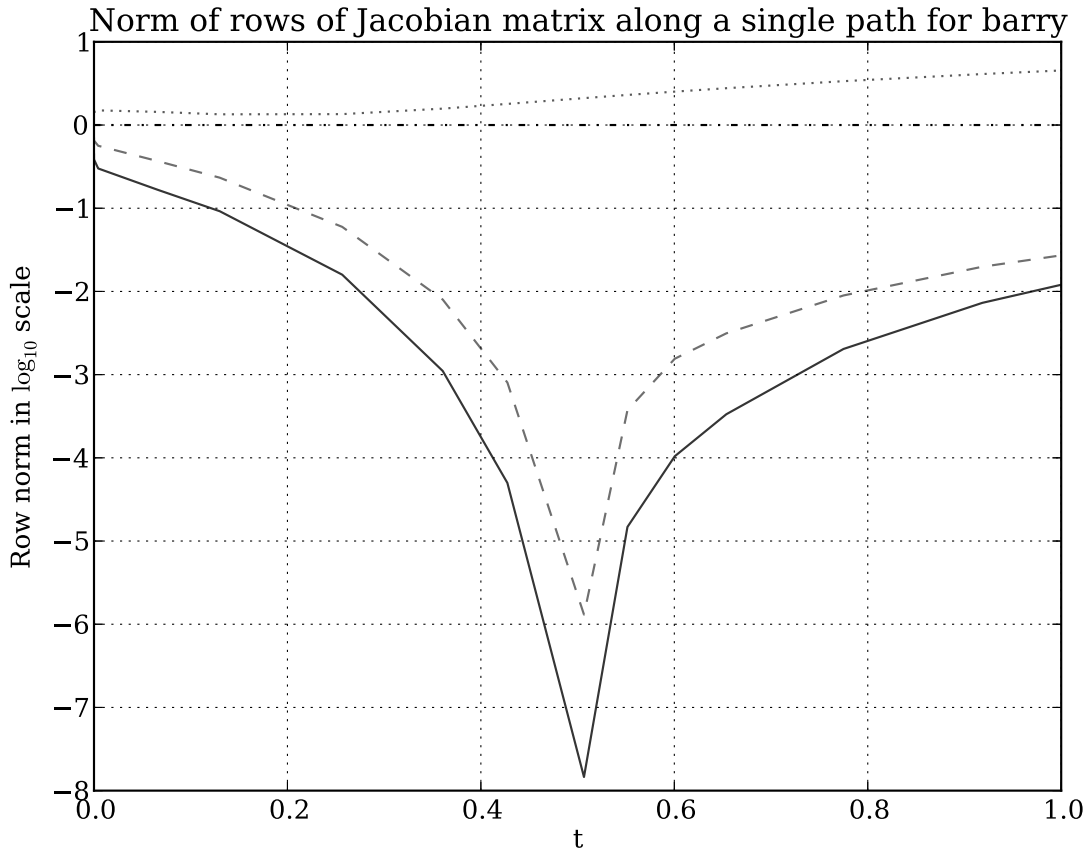


Figure 2.5: The 2-norm of rows, in $\log_{10}$ scale, of the Jacobian matrix along a single path defined by $\hat{H} = \mathbf{0}$, tracked using projective path tracking algorithms on S^{2n+1}/U for solving the barry problem.

2.8.2 The general problem

Our experiments have shown that the problem of large difference in the scale of rows in Jacobian matrix revealed by the above observation is widespread when projective path tracking is in use. Indeed, this problem has plagued not only the path tracking algorithms on S^{2n+1}/U we have proposed here but also any path tracking algorithms involving the homogenization of the homotopy including the method using affine chart described in Equation (2.2). To see why, consider a homogeneous polynomial $f \in \mathbb{C}[x_0, \dots, x_n]$ of degree d , then each $\frac{\partial f}{\partial x_j}$ for $j = 0, \dots, n$ is homogeneous of degree $d - 1$ unless f is constant respect to x_j . In either case, we have

$$\frac{\partial f}{\partial x_j}(\lambda x_0, \dots, \lambda x_n) = \lambda^{d-1} \frac{\partial f}{\partial x_j}.$$

For a fixed t , if we write the Jacobian matrix of $F(\mathbf{x}) = \hat{H}(\mathbf{x}, t)$ respect to $\mathbf{x}$ at a point $\mathbf{x}$

as the row matrix $J(\mathbf{x}) = \begin{pmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_n \end{pmatrix}$, since $F = (f_1, \dots, f_n)$ is homogeneous, we have

$$J(\lambda \mathbf{x}) = \begin{pmatrix} \lambda^{d_1-1} \mathbf{v}_1 \\ \vdots \\ \lambda^{d_n-1} \mathbf{v}_n \end{pmatrix},$$

where $d_i = \deg f_i$. Namely, when the given point $\mathbf{x}$ is scaled by a fixed factor, the rows in J are scaled by different factors determined by the degrees of the corresponding polynomials, then when the original system contains polynomials of very different degrees. The difference in the scales of the rows in $J(\mathbf{x})$ can be very sensitive to the scaling $\mathbf{x} \mapsto \lambda \mathbf{x}$.

2.8.3 Dynamic row-scaling

The problem stated above can be solved quite simply via row scaling. Clearly, the system of linear equations

$$J\mathbf{v} = \mathbf{b}$$

is equivalent to

$$AJ\mathbf{v} = A\mathbf{b}$$

whenever A is invertible. So to bring all the rows to more or less the same scale, we can take A to be the diagonal matrix

$$A = \begin{pmatrix} 1/\nu_1 & & & \\ & \ddots & & \\ & & 1/\nu_n & \\ & & & 1 \end{pmatrix}$$

where $\nu_j = \|J_j\|$ is the norm of the j -th row of the Jacobian matrix J . In principle any norm can be used here. Our actual implementation uses the ∞ -norm, for the ease of computation.

Figure 2.6 shows the path condition of the same path tracked for solving the `barry` problem with and without the dynamic row-scaling technique. The difference is day and night. In solving a large number of polynomial systems with projective path tracking, this simple technique is helpful, and, in certain cases, essential in improving the path conditions.

However, it is important to note that while row-scaling is useful in improving the path

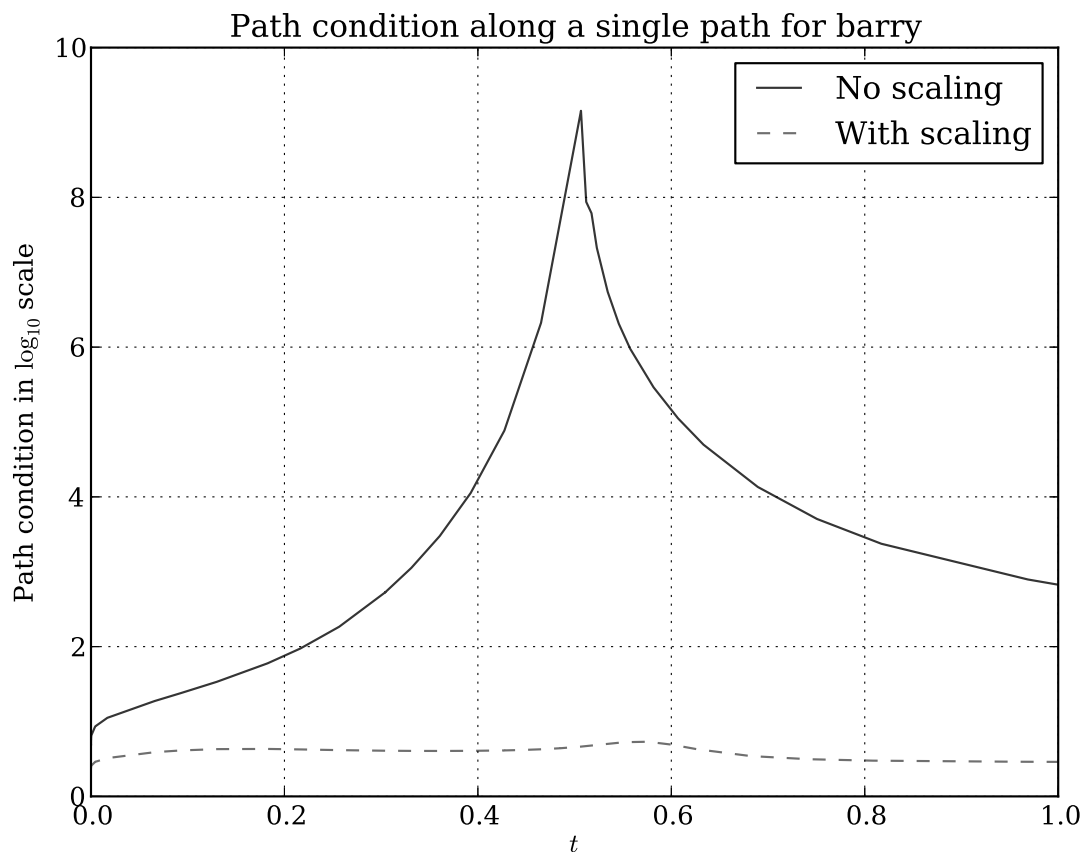


Figure 2.6: The comparison between path condition of a single path tracked in solving the **barry** problem using projective path tracking with (dashed) and without (solid) the dynamic row-scaling

condition, this transformation may conceal the fact that we are near a true singularity of a path, and thus it must be used with caution. Indeed this technique should only be used when one is confident that the path is away from a true singularity. In our actual implementation, this procedure is only active away from the endpoint at $t = 1$ where singularity may appear. Near the endpoint, the so called “endgame” techniques are used. Projective endgame is the topic of the next chapter.

2.9 The path tracking algorithm

The overall projective path tracking algorithm is summarized by the flowchart in Figure 2.7. In this flowchart, only the projective Newton's method is shown as the corrector, since it is indeed the main corrector we use in the path tracking algorithm. We should note, however, the other two projective correctors, the dampened projective Newton's method and the projective Newton's homotopy, can also replace or supplement the projective Newton's method. In this flowchart the operator $\mathcal{E}$ and $\mathcal{N}$ represent the projective Euler's method and the projective Newton's method respectively. $\epsilon_{\mathcal{N}}$ is the threshold for testing the convergence of the projective Newton's method. When the tangential residual is less than this threshold, the projective Newton's method is considered to have succeeded. The real number ϵ_d is another threshold that plays a similar role: When the Riemannian distance moved by a single Newton's iteration is less than this threshold, the projective Newton's method is also considered to have succeeded. $k_{\max}$ is the maximum number of Newton's iteration the corrector is allowed to use.

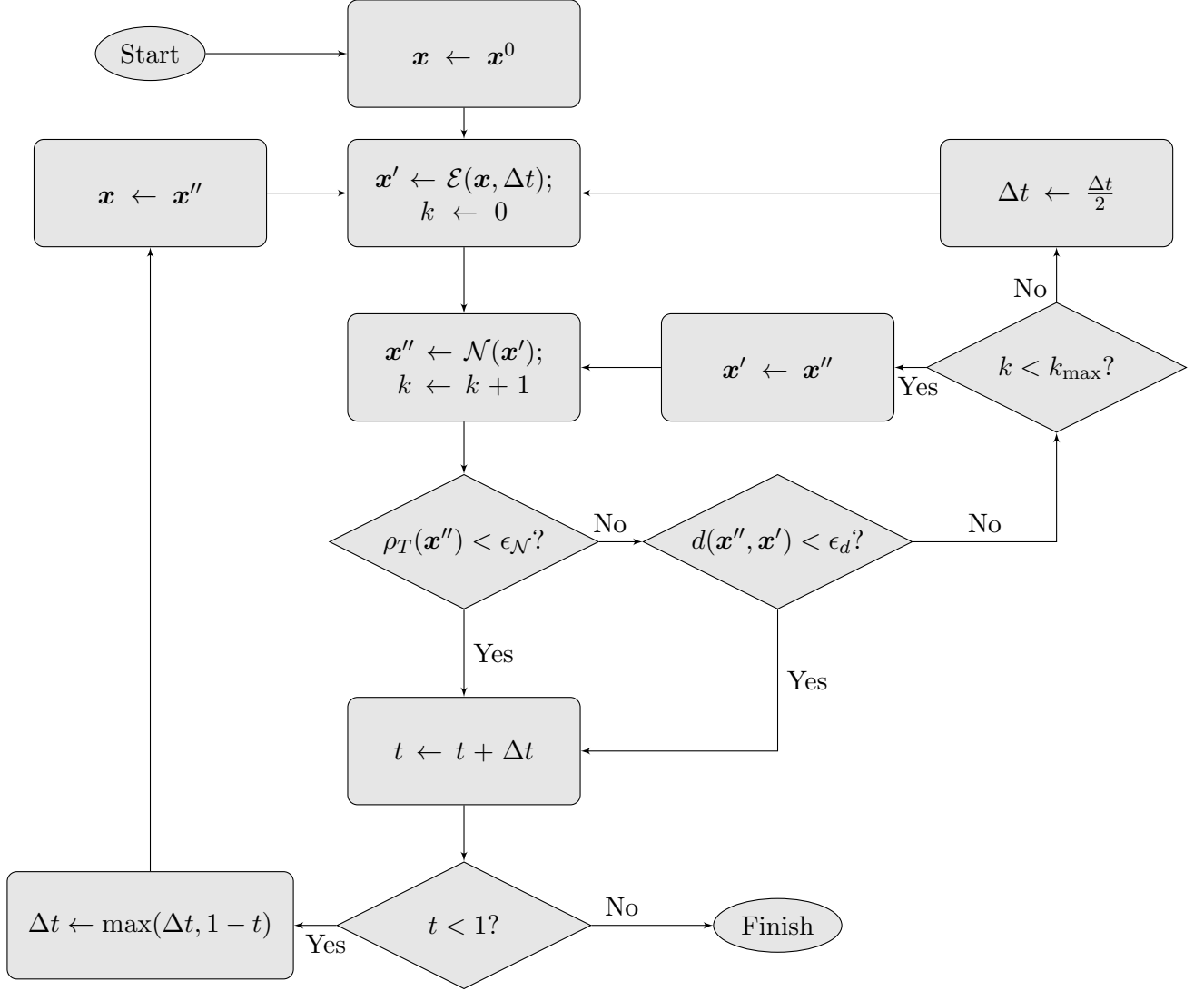


Figure 2.7: The flowchart summarizes the overall projective path tracking algorithm

2.10 Numerical results

In this section we present some numerical results obtained when the projective path tracking algorithms on S^{2n+1}/U is used. We mainly focus on the improvements on path conditions in contrast to the traditional approach of using generic hyperplane charts, described in Equation (2.2). Here we chose a well known problem: the `cyclic7` problem. However it is a representative of similar results we obtained from a large set of problems. Figure 2.8 shows the distribution of the maximum path conditions along each path tracked in solving the `cyclic7` problem using

the projective path tracking algorithms on S^{2n+1}/U with the dynamic row-scaling technique. Compared with the maximum path conditions when the method defined in Equation (2.2) is used, as shown in Figure 2.3, the paths here appear to be much more tamed with maximum condition number less than 10^4 which is well within the reach of double-precision floating point arithmetic.

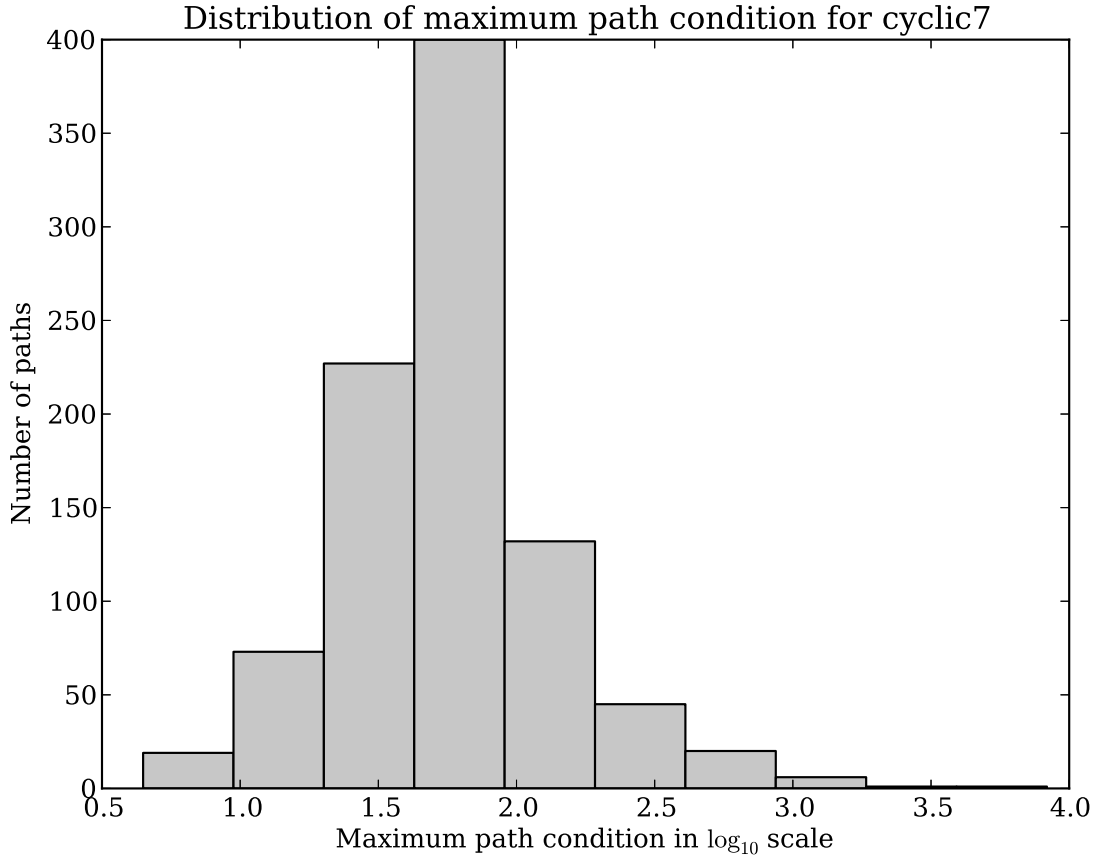


Figure 2.8: The histogram shows the distribution of maximum path conditions along all paths tracked for solving the **cyclic7** problem using projective path tracking algorithms on S^{2n+1}/U together with the dynamic row-scaling technique.

The difference is more obvious when we plot the distributions of maximum path conditions obtained from different path tracking methods. In Figure 2.9 we can clearly see the difference.

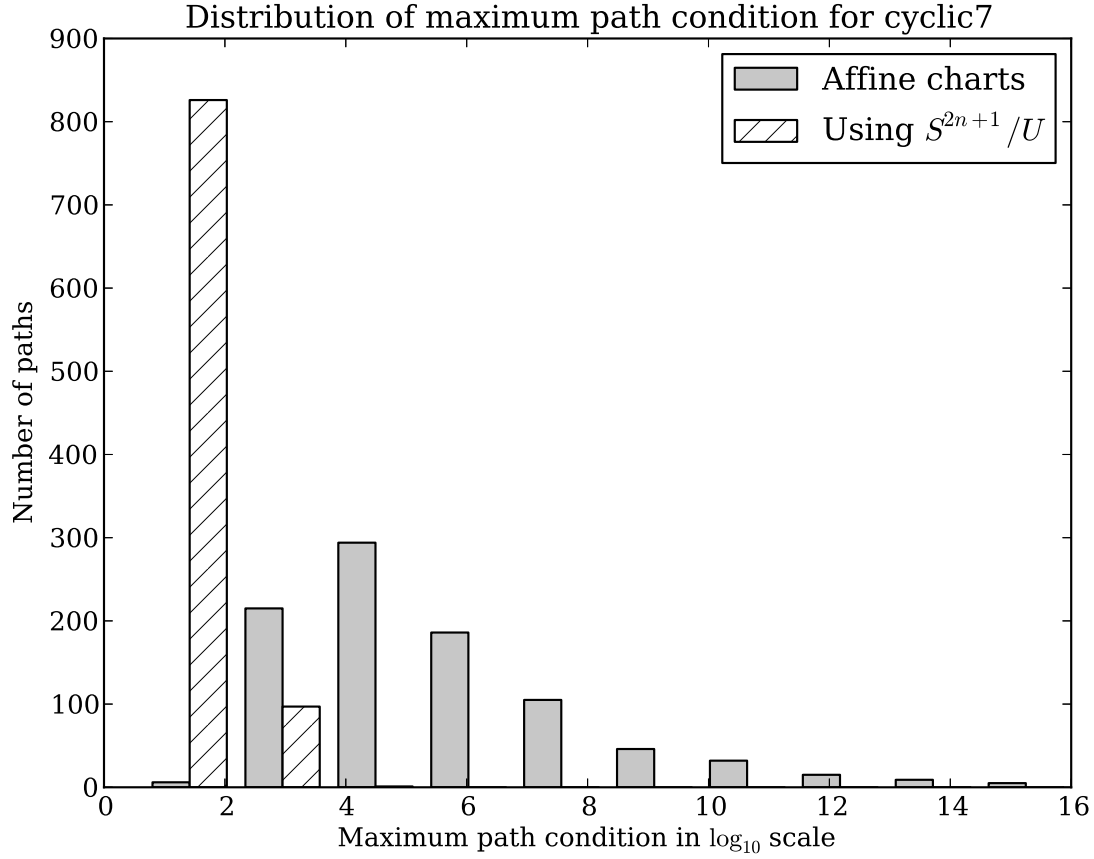


Figure 2.9: The histogram shows the distribution of maximum path conditions along all paths tracked for solving the `cyclic7` problem with the two different methods. The height of each solid bar represents the number of paths having the corresponding maximum path condition when the traditional method with affine charts described in Equation (2.2) is used. The striped bars show the same information when the projective path tracking algorithms on S^{2n+1}/U developed in this chapter is used. As with previous cases, this graph represents the average result of 10 different runs.

Chapter 3

Endgame

Since $\mathbb{CP}^n$ is compact, a projective path $\hat{\gamma} \subset \mathbb{CP}^n \times [0, 1]$ defined by $\hat{H}(\mathbf{x}, t) = \mathbf{0}$ necessarily has an end point in $\mathbb{CP}^n$ at $t = 1$. However, as discussed in the previous chapters, it is possible that the end point lies outside $U_0 = \{[x_0 : \cdots : x_n] \mid x_0 \neq 0\}$ which we identify with $\mathbb{C}^n$. We generally use the expression “solutions at infinity” to describe such end points, and these solutions are usually considered extraneous. To identify these extraneous “end points at infinity”, it seems that one may just simply marks all those with $x_0 = 0$ or sufficiently close to 0. However, Table 3.1 shows a very different reality. Contained in that table is the scale of $|x_0|$ at the end points of a few paths from different systems that are known to be “at infinity”. While we expect these values of $|x_0|$ to be exactly 0 or at least within machine epsilon from 0, we can see the scale of $|x_0|$ range from 10^{-4} to as large as 10^{-2} . Thus it is clearly impossible to classify them as end point “at infinity” based on those numbers alone.

The reason, as it turns out, is that these end points happen to be singular in the sense that the smoothness condition fails at those points. It is well known that if the end point is singular, ordinary predictor-corrector methods for path tracking, as described in the previous chapters are unable to locate the end points accurately.

The job of the endgame is twofold. First, it should identify paths with end points being “solutions at infinity” of the target system. Second, it has to obtain the end point accurately even if a path has a singular end point. In the case the end point is indeed singular, a

System	$\log_{10} x_0 $ of the end point
noon3 [23]	-4
reimer2 [2]	-4
reimer3 [2]	-3
reimer4 [2]	-2
cyclic5 [3]	-4
cyclic5 [3]	-2

Table 3.1: The scales of $|x_0|$ at end points of some paths that are known to be “at infinity”: These paths comes from solving systems listed on the first column using the projective path tracking we have discussed.

bonus feature for the endgame would be to discover additional information that describe the geometric structure around the singular end point. In this chapter, we shall develop the proper language in which the endgame can be discussed rigorously as well as the numerical techniques with which the endgame problem can be solved in practice.

3.1 Local theory of homotopy paths

In this section, for completeness, we shall briefly outline some relevant concepts and theorems in the local theory of holomorphic varieties (a.k.a. analytic sets). Detailed discussions can be found in [7], [8], and [9]. The theory provides us the proper language to discuss and describe the endgame techniques.

3.1.1 Background: local theory of holomorphic varieties

In this section we shall briefly review the basic theory of holomorphic functions and holomorphic varieties. The study of the function theory on $\mathbb{C}^n$ starts from its topology. Open polydiscs will be used as the basis of the topology. An **open polydisc** in $\mathbb{C}^n$ is a subset $\Delta(\mathbf{w}; \mathbf{r}) \subset \mathbb{C}^n$ of the form $\Delta(\mathbf{w}; \mathbf{r}) = \{(z_1, \dots, z_n) \in \mathbb{C}^n : |z_j - w_j| < r_j \text{ for } j = 1, \dots, n\}$ where $\mathbf{w} = (w_1, \dots, w_n) \in \mathbb{C}^n$ is the **center** and $\mathbf{r} = (r_1, \dots, r_n) \in \mathbb{R}^n$ is the **polyradius**. In this chapter a “neighborhood” of a point is always an open polydisc centered at that point.

Definition 1. *A complex-valued function f defined on an open subset $D \subset \mathbb{C}^n$ is called **holomorphic** in D if each point $\mathbf{w} = (w_1, \dots, w_n) \in D$ has an open neighborhood U in D such that the function f has a power series expansion*

$$f(\mathbf{z}) = \sum_{v_1, \dots, v_n=0}^{\infty} a_{v_1, \dots, v_n} (z_1 - w_1)^{v_1} \cdots (z_n - w_n)^{v_n}$$

which converges for all $\mathbf{z} = (z_1, \dots, z_n) \in U$.

In this chapter, we will focus on the local properties of holomorphic functions. We thus need the language of function germs.

Definition 2. *Two $\mathbb{C}$ -valued functions $f : U \rightarrow \mathbb{C}$ and $g : V \rightarrow \mathbb{C}$ are called **equivalent at the point $\mathbf{p} \in U \cap V$** if there is an open neighborhood W of $\mathbf{p}$ such that $W \subseteq U \cap V$ and $f \equiv g$ on W . This is an equivalence relation, and an equivalence class is called a **germ of $\mathbb{C}$ -valued function at the point $\mathbf{p}$** .*

The set of all such germs at a point $\mathbf{p} \in \mathbb{C}^n$ naturally forms a ring under the pointwise addition and multiplication. The field of complex numbers $\mathbb{C}$ sits inside this ring in the form of constant functions with $1 \in \mathbb{C}$ being the unity of the ring. Therefore the set of

germs actually has a $\mathbb{C}$ -algebra structure. Within this algebra, the set of germs given by holomorphic functions forms a sub $\mathbb{C}$ -algebra called the **germs of holomorphic functions**, for which we will use the notation ${}_n\mathcal{O}_{\mathbf{p}}$. It is clear that for any two points $\mathbf{p}, \mathbf{q} \in \mathbb{C}^n$, ${}_n\mathcal{O}_{\mathbf{p}}$ and ${}_n\mathcal{O}_{\mathbf{q}}$ are isomorphic as $\mathbb{C}$ -algebras. Therefore the local theory of holomorphic functions is the same at any point. It is easy to check that ${}_n\mathcal{O}_{\mathbf{p}}$ is an *integral domain* and hence has a well defined field of quotients ${}_n\mathcal{M}_{\mathbf{p}}$, the **field of germs of meromorphic functions**. It is also easy to see that ${}_n\mathcal{O}_{\mathbf{p}}$ is a *local ring* with non-units forming the unique maximal ideal. The Weierstrass Preparation Theorem [9, p.68 Theorem 2] and Weierstrass Division Theorem [9, p.70 Theorem 3] are two of the important theorems that reveal the structure of the embedding ${}_{n-1}\mathcal{O}_{\mathbf{p}} \subset {}_{n-1}\mathcal{O}_{\mathbf{p}}[z_n] \subset {}_n\mathcal{O}_{\mathbf{p}}$. They form the stepping stones of many induction proofs about the structure of ${}_n\mathcal{O}_{\mathbf{p}}$. In particular they are used to establish the fact that ${}_n\mathcal{O}_{\mathbf{p}}$ is *Noetherian* [8, Theorem 2, p.7], which is an extension of the Hilbert basis theorem. We shall now turn our attention to the zero sets of holomorphic functions.

Definition 3. Let B be an open subset of $\mathbb{C}^n$. A **holomorphic subvariety** of B is a subset V of B such that for each $\mathbf{p} \in V$ there exists a neighborhood U of $\mathbf{p}$ and $f_1, \dots, f_k$ holomorphic in U such that

$$V \cap U = \{z : f_1(z) = \dots = f_k(z) = 0\}.$$

Just like the situation for holomorphic functions, the language of germs facilitates our study of local behavior of holomorphic varieties:

Definition 4. For open sets B_1, B_2 in $\mathbb{C}^n$, if V_1 and V_2 are holomorphic subvarieties of B_1, B_2 that contain $\mathbf{p}$ respectively, we say V_1 is **equivalent to** V_2 at $\mathbf{p}$ if there exists a

neighborhood U of $\mathbf{p}$ on which

$$V_1 \cap U = V_2 \cap U.$$

It is easy to check that this indeed defines an equivalence relation, and hence we can talk about the equivalence classes of holomorphic subvarieties.

Definition 5. *A **germ at $\mathbf{p}$ of a holomorphic subvariety** is an equivalence class of holomorphic subvarieties under the above equivalence relation. We say a germ of holomorphic subvariety is **reducible** if it is the union of two proper germs of holomorphic subvarieties, and **irreducible** otherwise.*

Now we have two different kinds of objects. On the algebraic side, we have germs of holomorphic functions at a point. On the geometric side, we see germs of holomorphic subvarieties at a point. The relationship between these two kinds mimics that between polynomials and varieties that is central to algebraic geometry. For $f_1, \dots, f_k \in {}_n\mathcal{O}_{\mathbf{p}}$, we use the standard notation $\mathbb{V}(f_1, \dots, f_k)$ to denote the germ of holomorphic subvariety defined by $f_1, \dots, f_k$. To simplify the notations, here we generalize the notion of holomorphic subvariety to allow the empty set $\emptyset$ to be considered as a germ of holomorphic subvariety: $\mathbb{V}(f_1, \dots, f_k) = \emptyset$ if any $f_j(\mathbf{p}) \neq 0$. We can extend this notation to any ideal $I \subseteq {}_n\mathcal{O}_{\mathbf{p}}$. Since ${}_n\mathcal{O}_{\mathbf{p}}$ is Noetherian, I is generated by finitely many germs of holomorphic functions represented by $f_1, \dots, f_k$ with some sufficiently small common domain U containing $\mathbf{p}$, then their common zero set $V = \{\mathbf{z} \in U : f_1(\mathbf{z}) = \dots = f_k(\mathbf{z}) = 0\}$ represents a germ of holomorphic subvariety at $\mathbf{p}$. We define this germ to be $\mathbb{V}(I)$. From the opposite direction, given a germ V of a holomorphic subvariety at $\mathbf{p} \in \mathbb{C}^n$, we use the notation $\mathbb{I}(V)$ to denote the set of germs of holomorphic functions $f \in {}_n\mathcal{O}_{\mathbf{p}}$ that vanish on V . One can show that $\mathbb{I}(V)$ is an ideal,

commonly known as the **ideal of the germ** V . Indeed, $\mathbb{I}(V)$ is radical, i.e., $\sqrt{\mathbb{I}(V)} = \mathbb{I}(V)$. The well-definedness of both definitions are actually not completely trivial to verify. We refer to [5, p.77] for the proof and discussions of the properties of the two operators. An important consequence is that any germ V of a holomorphic subvariety can be written as a *finite* union of irreducible germs. This union, called the **irreducible decomposition** of the germ V is actually unique up to a permutation. We can show this by noticing the ideal $\mathbb{I}(V)$ of the germ V is an ideal in ${}_n\mathcal{O}_p$ which is Noetherian. The complete proof can be found in [8, p.15 Theorem 7].

We shall now focus on irreducible germs of holomorphic subvarieties. Our goal is to have a canonical form for all such germs. Recall that a mapping between two topological spaces is called **finite** if the inverse image of each point is a finite set, and is called **proper** if the inverse image of any compact set is also compact. Furthermore, a **covering map** is a surjective continuous map from a locally path-connected topological space to another topological space such that around each point in its image, there exists an open neighborhood whose inverse image under this map is a disjoint union of connected open subsets in the domain [14, p.278].

Definition 6. *A continuous, finite, proper, surjective map $\pi : V \rightarrow W$ between two second-countable Hausdorff spaces is called a **finite branched covering** if there is a dense open subsets $W_0 \subseteq W$ such that $V_0 = \pi^{-1}(W_0)$ is dense in V and the restriction of $\pi_0 : V_0 \rightarrow W_0$ of π on V_0 is a covering map.*

Here the space W is called the **base**, while V is called the **cover**. The restriction $\pi_0 : V_0 \rightarrow W_0$ of π on V_0 is called a **regular part** of the finite branched covering $\pi : V \rightarrow W$. So far, this construction is purely topological. What we need is an analytic version of this

concept:

Definition 7. A finite branched covering $\pi : V \rightarrow W$ between two holomorphic varieties, as topological spaces, is a **finite branched holomorphic covering** if there is a regular part $\pi_0 : V_0 \rightarrow W_0$ of π for which $W \setminus W_0$ is a holomorphic subvariety of W and π_0 is a locally biholomorphic mapping.

Theorem 6. (Local parametrization theorem) [8, Theorem 10, p.48] For an irreducible germ V at $\mathbf{p} \in \mathbb{C}^n$ of a holomorphic subvariety, with a suitable nonsingular linear change of coordinates of $\mathbb{C}^n$ there exists an integer $d \leq n$ such that the natural projection $\pi : \mathbb{C}^n = \mathbb{C}^d \times \mathbb{C}^{n-d} \rightarrow \mathbb{C}^d$, restricted on V , is a finite branched holomorphic covering over some open set in $\mathbb{C}^d$.

We call the smallest such integer d the **Weierstrass dimension**, or simply the **dimension**, of the irreducible germ V . Note that in this context, the regular part of this branched holomorphic covering π_V over some open set in $\mathbb{C}^d$ is a connected complex manifold of complex dimension d . For the rest of the discussion, we will only focus on the case when the dimension is 1, and in this case, we have a particularly simple and useful description of the cover. Notice that Theorem 6 is essentially a local theory, so we should be able to easily extend it into the projective space, after all $\mathbb{CP}^n$ is locally affine.

3.1.2 Local normal form of affine homotopy paths

Now we would like to derive the canonical form of a path $\gamma \subset \mathbb{C}^n \times [0, 1]$ defined by

$$H(\mathbf{x}, t) = \mathbf{0}$$

that converges to some end point $\zeta \in \mathbb{C}^n$ at $t = 1$. If we consider $H = (h_1, \dots, h_n)$ as a map $H : \mathbb{C}^{n+1} \rightarrow \mathbb{C}^n$ holomorphic in $(\mathbf{x}, t) = (x_1, \dots, x_n, t)$, then the zero set $\{(\mathbf{x}, t) \in \mathbb{C}^{n+1} \mid H(\mathbf{x}, t) = \mathbf{0}\}$ is a holomorphic subvariety of some domain in $\mathbb{C}^{n+1}$. We can then consider the germ of the holomorphic subvariety $V = \mathbb{V}(h_1, \dots, h_n)$ at the point $(\zeta, 1) \in \mathbb{C}^{n+1}$.

In this context the endgame in path tracking for homotopy continuation method can be understood as the geometric characterization of the germ V at the end point $(\zeta, 1)$ as well as the computation of an accurate estimate for the end point $(\zeta, 1)$ using these geometric information. To outline the basic idea, we shall start with a few nontrivial observations: Firstly, in the irreducible decomposition $V = V_1 \cup \dots \cup V_\ell$, over a sufficiently small t -interval the path γ must lie in exactly one such irreducible germ. Secondly, by Theorem 6 this irreducible germ can be realized as a finite branched holomorphic covering over some domain in $\mathbb{C}^d$ for which the regular part is a complex manifold of dimension d . But by the smoothness condition, we necessarily have $d = 1$. Thirdly, topologically speaking, the finite branched covering must be isomorphic to the standard finite branched covering given by $z \mapsto z^m$ where m is the number of sheets. Finally, the role of the special variable that serves as the base space can be played by none other than the path parameter t . These observations are made precise by the the following important theorem:

Theorem 7. *With the notation used above, at $t = 1$, the path γ has the convergent power series expansion of the form*

$$\begin{aligned} x_j &= \sum_{k=0}^{\infty} a_{jk} s^k \\ t &= 1 - s^m \end{aligned}$$

for $j = 1, \dots, n$, where $m \in \mathbb{Z}^+$.

We refer to [25] for the proof of this theorem based on the Local Parametrization Theorem 6. A number of other different approaches can be used to prove this theorem, in particular one also can consider it as the Local Normal Form theorem for holomorphic maps between Riemann surfaces, which can be found in [20]. A variety of endgames were developed based on this theorem.

3.1.3 Local normal form of projective homotopy paths

The goal of this section is to derive series expansions for projective paths similar to that provided by Theorem 7. These results are well developed, we refer to [20] for more in depth discussion. Let $\hat{\gamma} \subset \mathbb{CP}^n \times [0, 1]$ be a path defined by $\hat{H}(\mathbf{x}, t) = 0$ that has an affine associate $\gamma \subset \mathbb{C}^n \times [0, 1]$ which is defined for all $t \in (0, 1)$, i.e., $x_0 \neq 0$ along $\hat{\gamma}$ for $t \in (0, 1)$. By the smoothness condition, they are smooth paths parametrized by t for $t \in (0, 1)$ in $\mathbb{CP}^n$ and $\mathbb{C}^n$ respectively. Because $\mathbb{CP}^n$ is compact, $\hat{\gamma}$ necessarily converge to some point $[\zeta] = [\zeta_0 : \dots : \zeta_n]$ in $\mathbb{CP}^n$ at $t = 1$. By the previous analysis, the projection of the germ represented by $\hat{\gamma}$ at ζ onto the hyperplane

$$L(\mathbf{x}) := \langle \zeta, \mathbf{x} \rangle_{\mathbb{C}} - 1 = 0$$

where $\mathbf{x} = (x_0, \dots, x_n) \in \mathbb{C}^{n+1}$ is also smooth. This additional equation defines $\hat{\gamma}$ as an affine homotopy path to which Theorem 7 may be applied. Algebraically, there exists a punctured disk $D = \Delta(1; \varepsilon) \setminus \{1\}$ such that for each $t \in D$, there is a representative $(x_0, \dots, x_n) \in \mathbb{C}^{n+1}$

of $\hat{\gamma}(t)$ that is a nonsingular solution of the system

$$\hat{H}(x_0, \dots, x_n, t) = \mathbf{0}$$

$$L(x_0, \dots, x_n) = 0.$$

By Theorem 7, $t = 1 - s^m$ and $x_j = \sum_{k=0}^{\infty} a_{jk} s^k$ for $j = 0, \dots, n$. By assumption $\hat{\gamma}$ has an affine associate $\gamma \subset \mathbb{C}^n \times [0, 1]$ which is a smooth path in $\mathbb{C}^n$ parametrized by t for $t \in (0, 1)$.

So it is necessary that $x_0 \neq 0 \in {}_1\mathcal{O}_0$, i.e., it is not the zero power series. To summarize, near $t = 1$, the projective path $\hat{\gamma}$ can be parametrized by a holomorphic map in a variable s of the form

$$\left\{ \begin{array}{lcl} x_0 & = & \sum_{k=0}^{\infty} a_{0k} s^k \neq 0 \\ x_1 & = & \sum_{k=0}^{\infty} a_{1k} s^k \\ & \vdots & \\ x_n & = & \sum_{k=0}^{\infty} a_{nk} s^k \\ t(s) & = & 1 - s^m \end{array} \right. \quad (3.1)$$

which satisfies the additional equations

$$a_{00} = \zeta_0$$

$$\vdots \quad \vdots \quad \vdots$$

$$a_{n0} = \zeta_n$$

$$\langle \zeta, \mathbf{x}(s) \rangle = 1.$$

Remark 1. *In light of Lemma 1 and the analysis in Section 2.1.2, the role of the true end point ζ can be played by some sufficiently close $\tilde{\zeta}$ in this power series expansion. Of course,*

it is generally impossible to determine, a priori, how close is sufficiently close.

Note that such a power series expansion exists for paths that converges to point in $\mathbb{C}^n$ or outside $\mathbb{C}^n$ (points at “infinity”). While it is dependent on the hyperplane defined by $L(\mathbf{x}) = \langle \boldsymbol{\zeta}, \mathbf{x} \rangle - 1 = 0$, we can easily remove this artificial dependency by rewriting the series and obtain a convenient alternative expansion. Since $x_0(s)$ is not the zero power series in s , it must be of finite order. Let d be its order, then

$$x_0 = s^d \sum_{k=0}^{\infty} a_{0k} s^k$$

for $a_{00} \neq 0$. Then $\sum_{k=0}^{\infty} a_{0k} s^k$ is order 0 and hence invertible in the ring ${}_1\mathcal{O}_0$. Let $b \in {}_1\mathcal{O}_0$ be its inverse, then in ${}_1\mathcal{M}_0$, the inverse of x_0 has the form $s^{-d}b$. Since the affine associate γ of $\hat{\gamma}$ is given by $\left(\frac{x_1}{x_0}, \dots, \frac{x_n}{x_0}, t\right)$ for $([x_0 : \dots : x_n], t) \in \hat{\gamma}$, each coordinate $\frac{x_j}{x_0}$ can be formally written as

$$\frac{x_j}{x_0} = s^{-d}b \sum_{k=0}^{\infty} a_{jk} s^k = s^{-d} \sum_{k=0}^{\infty} \tilde{a}_{jk} s^k = \sum_{k=-d}^{\infty} \tilde{a}_{jk} s^k$$

which are well defined germs of meromorphic functions in ${}_1\mathcal{M}_0$. Therefore in affine space we have Laurent series expansions of the form

$$\begin{aligned} x_j &= \sum_{k=-d}^{\infty} a_{jk} s^k \\ t &= 1 - s^m \end{aligned}$$

for some finite $d \in \mathbb{Z}$ with $j = 1, \dots, n$. By “factoring” out the lowest term from the series expansion for each x_j , we can write x_j as the product of $a_{j,-d} s^{-d} \in {}_1\mathcal{M}_0$ and another series

in ${}_1\mathcal{O}_0$. After relabeling, we obtain an equivalent form in the affine space $\mathbb{C}^m$:

$$\left\{ \begin{array}{lcl} x_1 & = & a_1 s^{\omega_1} \left(1 + \sum_{k=1}^{\infty} a_{1k} s^k \right) \\ & \vdots & \\ x_n & = & a_n s^{\omega_n} \left(1 + \sum_{k=1}^{\infty} a_{nk} s^k \right) \\ t(s) & = & 1 - s^m \end{array} \right. \quad (3.2)$$

where $\omega_1, \dots, \omega_n \in \mathbb{Z}$ are the orders of the power series expansions.

The power series expansions (3.1) and (3.2) allow us to perform endgame classification via well developed power series techniques (a.k.a., Puiseux series techniques) described in [11],[15], [17], and [25] in conjunction with projective path tracking. These techniques have been used in our actual implementation, and they are efficient and effective in many cases. However, as explicitly pointed out by [25, p.186] and [11], their usefulness diminish as the integer m increases. Indeed our experiments have shown that it is generally of little hope in applying these techniques to paths with m greater than three or four using standard double-precision floating point arithmetic only. Thus the power series techniques are only used as the first line of defense in our implementation. A more powerful, albeit expensive, technique via Cauchy integral is preferred.

3.2 Projective endgame based on Cauchy integral

Let us continue to use the notations $\hat{\gamma}$ and γ for the projective path and its affine associate with $([\zeta], 1)$ being the end point of $\hat{\gamma}$. With (3.1), the endgame based on Cauchy integral described in [25] can be easily extended to $\mathbb{CP}^n$. Let $([\mathbf{x}^0], t_0)$ be a point on $\hat{\gamma}$ sufficiently close to $([\zeta], 1)$. By (3.1) and Lemma 1 the coordinates of the projection of the path $\hat{\gamma}$ to the hyperplane defined by $L(\mathbf{x}) = \langle \mathbf{x}^0, \mathbf{x} \rangle_{\mathbb{C}} - 1 = 0$ is parametrized by a holomorphic map given by $x_j(s) = \sum_{k=0}^{\infty} a_{0k} s^k$ for each $j = 0, \dots, n$. By the Cauchy Integral Theorem,

$$x_j(0) = \frac{1}{2\pi i} \int_{\Gamma} \frac{x_j(s)}{s} ds$$

for each $j = 0, \dots, n$ and any loop Γ around $s = 0$ which is the only possible singularity of $x_j(s)$ in the interior of Γ . More generally, written in a vector notation, we have

$$\mathbf{x}(0) = \frac{1}{2\pi i} \int_{\Gamma} \frac{\mathbf{x}(s)}{s} ds. \quad (3.3)$$

In other words, the value of $\mathbf{x}$ at the end point of the path $\hat{\gamma}$ can be computed via an integral, since $[\zeta] = [\mathbf{x}(0)]$ as points in $\mathbb{CP}^n$. This is the basic idea behind the projective Cauchy integral endgame. Unfortunately, this integral cannot be computed directly since the relationship between t and s is not known. Thus we first need to know the value of m in $t = 1 - s^m$. Second, we need to collect numerical samples of the value of $\mathbf{x}$ along the loop Γ , so that the above integral can be evaluated. Conveniently, the two pieces of information can be obtained from the very same process, killing two birds with one stone.

Recall that $t = 1 - s^m$ which may not have holomorphic inverse in a disk centered at $s = 0$. However, we certainly have a holomorphic branch of the “inverse”, given by $s = e^{\frac{1}{m} \log(1-t)}$

with the principle branch of the logarithm function. Then along this branch, the value of $\mathbf{x}$ can be expressed as a function of t given by $\mathbf{x} \left(e^{\frac{1}{m} \log(1-t)} \right)$. If we parametrize the small circle of radius $r = 1 - t_0$ centered at 1 in t -space as $t = 1 - re^{i\theta}$ using the angle θ , the value of $\mathbf{x}$, now as a function of θ , can be expressed as

$$\phi(\theta) := \mathbf{x} \left(e^{\frac{1}{m} \log(re^{i\theta})} \right) = \mathbf{x} \left(\sqrt[m]{r} e^{i\theta/m} \right).$$

Then it is easy to check that m is the smallest natural number among all $k \in \mathbb{Z}^+$ such that $\phi(2k\pi) = \phi(0)$. That is, the values $\Phi = \{\phi(\theta) \mid \theta \in \mathbb{R}\} \subset \mathbb{CP}^n$ form a (closed) loop parametrized by $\theta \in [0, 2m\pi]$.

In the projective Cauchy integral endgame, we consider the loop Φ as the projection on the $\mathbf{x}$ -space of a homotopy path defined by the equation

$$\hat{H}(\mathbf{x}, 1 - re^{i\theta}) \equiv 0.$$

Hence the techniques described in the previous chapter can be used to track the path Φ . The strategy is then to track the value of $\phi(\theta)$ as θ increases in discrete steps until the first occurrence of $\phi(2k\pi) = \phi(0)$, at which point we will obtain two things: The value k which gives us m and the set of sample values of $\phi(\theta)$ with which the above integral can be approximated numerically, which in turn gives us the estimate of the end point $([\zeta], 1)$.

More precisely, the process has three stages: the sampling stage, integration stage, and the verification stage. In the sampling stage, we perform the projective path tracking algorithms to track the path $\phi(\theta)$ in prescribed and equally spaced steps $\theta_1 = \Delta\theta, \theta_2 = 2\Delta\theta, \dots$ starting from $\theta_0 = 0$ and the initial point $\phi(0) = \mathbf{x}^0$. During this path tracking process, the model

$\mathbb{CP}^n = S^{2n+1}/U$ is used, so all the sample points $\phi(\theta_j)$ have unit norm. The sample points $\phi(\theta_1), \phi(\theta_2), \dots$ are then scaled via

$$\hat{\phi}(\theta_j) := \frac{\phi(\theta_j)}{\langle \mathbf{x}^0, \phi(\theta_j) \rangle_{\mathbb{C}}}$$

so that they lie on the hyperplane defined by $L(\mathbf{x}) = \langle \mathbf{x}^0, \mathbf{x} \rangle_{\mathbb{C}} - 1 = 0$. This process continues until we have determined that $\phi(\theta_j) = \phi(0)$ for some $\theta_\ell = 2k\pi$ where $k \in \mathbb{Z}^+$ is the **winding number** of the loop. This stopping criterion is, of course, an ill-posed question. We will discuss the stopping criteria in the following subsections. Next, the samples are used to approximate the Cauchy integral 3.3 using the trapezoid method given by

$$\tilde{\zeta} = \frac{1}{\ell} \sum_{j=0}^{\ell} \phi(\theta_j)$$

Finally, in the verification stage the residual $\rho(\tilde{\zeta})$ is computed. Since we expect $\tilde{\zeta}$ to be a close approximation of a solution to $\hat{H}(\mathbf{x}, 1) = 0$, the residual $\rho(\tilde{\zeta})$ should be relatively small. If $\rho(\tilde{\zeta})$ is greater than a certain threshold, the result is discarded, and we consider the Cauchy integral endgame to have failed. From our experiences, this criterion is generally sufficient to verify the projective Cauchy integral endgame has worked correctly. However, in Section 3.2.3, we will discuss a stronger verification process.

It is experimentally verified that when the projective Cauchy integral endgame works correctly, it provides us *much better accuracy* than what can be obtained from projective path tracking algorithm alone for singular endpoints. Using the Total Degree Homotopy [16] with projective path tracking developed in the previous chapter, all paths of `noon3`, `reimer2`, `reimer3`, `cyclic4`, and `cyclic5` can be tracked to their endpoints. Unfortunately, as shown

in Figure 3.1, in most cases the scale of $|x_0|$ makes a poor indicator of the path going to “infinity”. With projective Cauchy integral endgame, however, we can improve the estimate of x_0 greatly: the aforementioned paths now has $|x_0|$ at end points close to machine epsilon using double-precision floating point arithmetic, making it a good indication that the path end point is “at infinity”.

System	Winding number	$\log_{10} x_0 $	
		with path tracking	with Cauchy integral
noon3	2	−4	−17
reimer2	2	−4	−16
reimer3	3	−3	−15
reimer4	4	−2	−14
cyclic5	2	−4	−15
cyclic5	5	−2	−15

Table 3.2: Comparison of the magnitude of $|x_0|$, in $\log_{10}$ scale, with and without the Cauchy integral method. The second column shows the winding number of the loop used for evaluating Cauchy integral. The third column shows the approximate order of magnitude of $|x_0|$ at the end point using the projective path tracking alone, while the last column shows the same measure but with Cauchy integral producing the end point.

3.2.1 Stopping criteria based on Riemannian distance

When do we stop collecting samples of $\phi(\theta_j)$? Namely, how many $\{\phi(\theta_0), \phi(\theta_1), \dots\}$ should we collect? In theory, we should stop at the smallest natural number k such that $[\phi(2k\pi)] = [\phi(0)]$ as points in $\mathbb{CP}^n$, and for this purpose the Riemannian distance function $d = d_{\mathbb{CP}^n}$ on $\mathbb{CP}^n$ can be used, i.e., we require

$$d([\phi(2k\pi)], [\phi(0)]) = 0.$$

However, with numerical error, this criterion is unlikely to ever be satisfied exactly. So we must answer the question: how small is small enough? From the experiments, we found

it doubtful that there is a threshold one can use for all systems. In general the distance $d([\phi(2k\pi)], [\phi(0)])$ must be compared to the size of the loop Φ . So we can keep track of the maximum distance from $[\phi(0)]$ to points $[\phi(\theta_1)], [\phi(\theta_2)], \dots$

$$d_{\max} := \max_j d([\phi(\theta_j)], [\phi(0)])$$

which should provide us a good estimation of the size of the loop Φ . Then the closing of the loop can be numerically determined by the condition

$$\frac{d([\phi(2k\pi)], [\phi(0)])}{d_{\max}} < \varepsilon$$

for certain threshold ε . In our preliminary implementation, the value $\varepsilon = 10^{-3}$ is used, and it is successful as a stopping criteria for almost all paths with a winding number 12 or less.

3.2.2 Stopping criteria based on tangent vector

A potentially more robust stopping criteria comes from the observation that if Φ does indeed form a loop with k being the smallest natural number such that $\phi(2k\pi) = \phi(0)$, then it is, in addition, a smooth loop, which means as tangent vectors of $T_{[\phi(0)]}\mathbb{CP}^n$ we must have

$$\dot{\phi}(2k\pi) = \dot{\phi}(0),$$

therefore the angle between the two vectors

$$\cos^{-1} \frac{\langle \dot{\phi}(2k\pi), \dot{\phi}(0) \rangle_{\mathbb{R}}}{\|\dot{\phi}(2k\pi)\|_2 \cdot \|\dot{\phi}(0)\|_2}$$

should give us another indicator for whether or not we have collected a full loop. Together with the stopping criteria based on the Riemannian distance this should provide us a more robust criteria, in theory at least. In our numerical experiments, we found this technique being helpful in preventing premature termination of the sampling stage in certain cases.

3.2.3 Consistency tests

As a final line of defense, the projective Cauchy integral endgame is computed with smaller and smaller radius $r = 1 - t_0$ closer and closer to the end point at $t = 1$. If each iteration worked correctly, we would expect

- The winding number $m^1, m^2, \dots$ remain a constant,
- The variation of the endpoint estimates $\tilde{\zeta}^1, \tilde{\zeta}^2, \dots$ should be no greater than twice the square root of the machine epsilon.

Thus the stabilizing of the results of successive projective Cauchy integral endgames should give us a very strong indication that the projective Cauchy integral has worked correctly and provided us the accurate approximation of the end point $\tilde{\zeta} \approx \zeta$.

BIBLIOGRAPHY

BIBLIOGRAPHY

- [1] E.L. Allgower and K. Georg. *Introduction to numerical continuation methods*, volume 45. Society for Industrial and Applied Mathematics, 2003.
- [2] G. Attardi and C. Traverso. The PoSSo library for polynomial system solving. *Proc. of AIHENP95*, 1995.
- [3] G. Bjorck and R. Froberg. A faster way to count the solutions of inhomogeneous systems of algebraic equations, with applications to cyclic n -roots. *Journal of Symbolic Computation*, 12(3):329–336, 1991.
- [4] L. Blum, Felipe Cucker, M. Shub, and S. Smale. *Complexity and real computation*. Springer-Verlag, 1998.
- [5] W. Ebeling. *Functions of several complex variables and their singularities*, volume 83. American Mathematical Society, 2007.
- [6] S. Gallot, D. Hulin, and J. Lafontaine. *Riemannian geometry*. Springer-Verlag, 2004.
- [7] R.C. Gunning. *Introduction to holomorphic functions of several variables: Function theory*, volume 1. Brooks/Cole Publishing Company, 1990.
- [8] R.C. Gunning. *Introduction to holomorphic functions of several variables: Local theory*, volume 2. Brooks/Cole Publishing Company, 1990.
- [9] R.C. Gunning and H. Rossi. *Analytic functions of several complex variables*. AMS Chelsea Publishing, 2009.
- [10] B. Huber and B. Sturmfels. A polyhedral method for solving sparse polynomial systems. *Mathematics of computation*, 64(212):1541–1555, 1995.
- [11] B. Huber and J. Verschelde. Polyhedral end games for polynomial continuation. *Numerical Algorithms*, 18(1):91–108, 1998.
- [12] J.M. Lee. *Riemannian manifolds: An introduction to curvature*, volume 176. Springer-Verlag, 1997.

- [13] J.M. Lee. *Introduction to smooth manifolds*, volume 218. Springer-Verlag, 2003.
- [14] J.M. Lee. *Introduction to topological manifolds*, volume 202. Springer-Verlag, 2011.
- [15] T.L. Lee, T.Y. Li, and C.H. Tsai. HOM4PS-2.0: a software package for solving polynomial systems by the polyhedral homotopy continuation method. *Computing*, 83(2):109–133, 2008.
- [16] T.Y. Li. On Chow, Mallet-Paret and Yorke homotopy for solving systems of polynomials. *Bulletin of the Institute of Mathematics. Acad. Sinica*, pages 433–437, 1983.
- [17] T.Y. Li. Numerical solution of polynomial systems by homotopy continuation methods. *Handbook of numerical analysis*, 11:209–304, 2003.
- [18] T.Y. Li, T. Sauer, and J.A. Yorke. The random product homotopy and deficient polynomial systems. *Numerische Mathematik*, 51(5):481–500, 1987.
- [19] T.Y. Li, T. Sauer, and J.A. Yorke. The cheater’s homotopy: an efficient procedure for solving systems of polynomial equations. *SIAM Journal on Numerical Analysis*, pages 1241–1251, 1989.
- [20] R. Miranda. *Algebraic curves and Riemann surfaces*, volume 5. American Mathematical Society, 1995.
- [21] A.P. Morgan. *Solving polynomial systems using continuation for engineering and scientific problems*, volume 57 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics, 2009.
- [22] A.P. Morgan and A.J. Sommese. A homotopy for solving general polynomial systems that respect m -homogeneous structures. *Applied Mathematics and Computation*, 24(2):101–113, 1987.
- [23] V.W. Noonburg. A neural network modeled by an adaptive Lotka-Volterra system. *SIAM Journal on Applied Mathematics*, pages 1779–1792, 1989.
- [24] M. Shub and S. Smale. On the complexity of Bezouts theorem I - geometric aspects. *Journal of the AMS*, 6(2), 1993.
- [25] A.J. Sommese and C.W. Wampler. *The Numerical solution of systems of polynomials arising in engineering and science*. World Scientific Pub Co Inc, 2005.

- [26] J. Stoer and R. Bulirsch. *Introduction to numerical analysis*, volume 12. Springer-Verlag, 2002.
- [27] G. Strang. *Linear algebra and its application*. Academic Press Inc., 1976.
- [28] D. Varolin. *Riemann surfaces by way of complex analytic geometry*, volume 125. American Mathematical Society, 2011.
- [29] R.J. Walker. *Algebraic curves*. Springer-Verlag, 1978.
- [30] C.W. Wampler, A.P. Morgan, and A.J. Sommese. Complete solution of the nine-point path synthesis problem for four-bar linkages. *Journal of Mechanical Design*, 114:153, 1992.
- [31] J.H. Wilkinson. *The algebraic eigenvalue problem*. Oxford University Press, USA, 1988.