

ABSTRACT

Title of dissertation: RECONSTRUCTION, ANALYSIS AND SYNTHESIS OF COLLECTIVE MOTION

Biswadip Dey, Doctor of Philosophy, 2015

Dissertation directed by: Professor P. S. Krishnaprasad
Dept. of Electrical & Computer Engineering

As collective motion plays a crucial role in modern day robotics and engineering, it seems appealing to seek inspiration from nature, which abounds with examples of collective motion (starling flocks, fish schools *etc.*). This approach towards understanding and reverse-engineering a particular aspect of nature forms the foundation of this dissertation, and its main contribution is threefold.

First we identify the importance of appropriate algorithms to extract parameters of motion from sampled observations of the trajectory, and then by assuming an appropriate generative model we turn this into a regularized inversion problem with the regularization term imposing smoothness of the reconstructed trajectory. First we assume a linear triple-integrator model, and by penalizing high values of the jerk path integral we reconstruct the trajectory through an analytical approach. Alternatively, the evolution of a trajectory can be governed by natural Frenet frame equations. Inadequacy of integrability theory for nonlinear systems poses the utmost challenge in having an analytic solution, and forces us to adopt a numerical optimization approach. However, by noting the fact that the underlying dynamics defines a left invariant vector field on a Lie group, we develop a framework based on Pontryagin's maximum principle. This approach toward data smoothing yields a semi-analytic solution.

Equipped with appropriate algorithms for trajectory reconstruction we analyze flight data for biological motions, and this marks the second contribution of this dissertation. By analyzing the flight data of big brown bats in two different settings (chasing a free-flying praying mantis and competing with a conspecific to catch a tethered mealworm), we provide evidence to show the presence of a context specific switch in flight strategy. Moreover, our approach provides a way to estimate the behavioral latency associated with these foraging behaviors. On the other hand, we have also analyzed the flight data of European starling flocks, and it can be concluded from our analysis that the flock-averaged coherence (the average cosine of the angle between the velocities of a focal bird and its neighborhood center of mass, averaged over the entire flock) gets maximized by considering 5-7 nearest neighbors. The analysis also sheds some light into the underlying feedback mechanism for steering control.

The third and final contribution of this dissertation lies in the domain of control law synthesis. Drawing inspiration from coherent movement of starling flocks, we introduce a strategy (*Topological Velocity Alignment*) for collective motion, wherein each agent aligns its velocity along the direction of motion of its neighborhood center of mass. A feedback law has also been proposed for achieving this strategy, and we have analyzed two special cases (two-body system; and an N -body system with cyclic interaction) to show effectiveness of our proposed feedback law. It has been observed through numerical simulation and robotic implementation that this approach towards collective motion can give rise to a splitting behavior.

RECONSTRUCTION, ANALYSIS AND SYNTHESIS
OF COLLECTIVE MOTION

by

Biswadip Dey

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2015

Advisory Committee:

Professor P. S. Krishnaprasad, Chair/Advisor

Professor S. I. Marcus

Professor A. L. Tits

Dr. E. W. Justh

Professor D. Levy, Dean's Representative

© Copyright by
Biswadip Dey
2015

Dedicated to Shawon and my parents

Acknowledgments

First and foremost, I would like to thank my advisor, Prof. P. S. Krishnaprasad for giving me an invaluable opportunity to work on challenging and extremely interesting problems over the past five and half years. He has always made himself available for help and advice, and there has never been an occasion when I have visited his office and he has not given me time. His attention to detail and dedication to exceptional scientific writing have made each of our manuscripts a work of art. Furthermore, his standards of excellence in research and teaching, and his passion for science, have been a continuous source of inspiration. It has been a pleasure to work with and learn from such an extraordinary individual.

I would also like to thank Prof. Steve Marcus, Prof. Andre Tits, Dr. Eric Justh and Prof. Doron Levy for agreeing to serve on my dissertation committee and for sparing their invaluable time reviewing the manuscript. I am particularly thankful to Prof. Tits for providing me guidance in improving the quality of Chapter 4.

I am indebted to the research group of Prof. Andrea Cavagna and Prof. Cynthia Moss, for sharing with me their experimental data on starling murmuration and bat foraging. These data formed the foundation for our analysis (in Chapter 5 and 6) on flight strategies and steering control mechanism.

This acknowledgement will be incomplete without recognizing the support of my teachers from the Bachelor's and Master's program at Jadavpur University and

IIT Bombay, respectively. In particular, Prof. Tapan Kumar Ghoshal, Prof. Ravi Banavar, Prof. Madhu Belur, Prof. Harish Pillai have played very instrumental roles in shaping my interest in control theory and motivating me towards research.

My friends and colleagues at the Intelligent Servosystems Laboratory have enriched my graduate life in many ways and they deserve a special mention. I am particularly thankful to Kevin Galloway, Matteo Mischiati, Ben Flom, Yunlong Huang, Vidya Raju, Udit Halder for all the stimulating discussions and feedback. It has been a pleasure to collaborate with Kevin and Udit during the last leg of this journey.

I would like to take this opportunity to acknowledge help and support from the staff members of the Electrical and Computer Engineering Department and the Institute for Systems Research. In particular, I would like to thank Ms. Pamela White, Ms. Alexis Jenkins, Ms. Regina King, Ms. Beverly Dennis and Ms. Melanie Prange. With Pam and Beverly's assistance, I never had to think about my conference registrations and travel arrangements. Also, the technical support from the ECE Department computing facilities help desk is highly appreciated.

Many other friends outside the lab have been great sources of conversation, inspiration, motivation and sometimes just pure fun. Agniv, Anup, Arijit, Arya, Barna, Dipankar, Jiaul, Pritam, Rajibul, Ranchu, Ranjith, Shinkyu, Siddharth, Soumyadip, Srimoyee and Udit - thank you all for making my time at College Park such a memorable one. I am also fortunate for my enduring friendships with Mrinmoy and Trishit, the best buddies one can ever ask for.

No accomplishment of mine has been made without the strong support of

family. My parents are a source of inspiration and have always done everything in their power to make my dreams a reality. My wife Shawon has been a great companion on this journey, lovingly sharing with me all the sacrifices, bitterness and happiness of our lives, and playing a leading role in sorting out our very own 2-body problem. I am grateful for all her support, care and love.

I gratefully acknowledge the financial supports from the Air Force Office of Scientific Research under AFOSR Grant No. FA9550-10-1-0250 and FA2386-12-1-3002 (FY2012 DURIP); from the Office of Naval Research under ODDR&E MURI Program Grant No. N000140710734; from the Army Research Laboratory under ARL/ARO MURI Program Grant No. W911NF-13-1-0390; and from the Kulkarni Foundation Summer Research Fellowship.

Table of Contents

1	Introduction	1
1.1	Background and Motivation	1
1.2	Thesis Outline	7
1.3	Mathematical Preliminaries	9
1.3.1	Self-steering Particle Model	9
1.3.2	Pursuit Strategies and Feedback Laws	12
I	Reconstruction of Collectives	18
2	Data Smoothing through Nonlinear Optimization - Mathematical Programming	19
2.1	Regularized Inversion Problem	22
2.2	Some Practical Issues	24
2.2.1	Numerical Integration for the Group Dynamics	24
2.2.2	Parametrization of $SO(3)$ Using Cayley Transform	27
2.2.3	Customization for Mathematical Programming	29
2.2.4	Multi-stage Approach for Optimization	31
2.3	Ordinary Cross Validation for the Regularized Inversion Problem	33
2.4	Numerical Results	35
2.5	Conclusion	36

3	Data Smoothing through Linear Quadratic Optimal Control	38
3.1	Problem Formulation	39
3.2	A Control Theoretic Approach	42
3.2.1	Path Independence Lemma and Its Application to Data Smoothing	43
3.2.2	Linearity of Reconstruction	59
3.3	An Alternative Co-state Based Approach	62
3.4	Ordinary Cross Validation for Optimal λ Selection	67
3.5	Numerical Results	73
3.6	Conclusion	74
4	Data Smoothing through Nonlinear Optimization - Maximum Principle	77
4.1	Data smoothing in a Euclidean setting	78
4.2	Data smoothing problems in a Finite Dimensional Matrix Lie group setting	93
4.3	A Quick Revisit to Lie-Poisson Reduction	99
4.4	Example I: Data Smoothing on $SO(3)$	102
4.4.1	Maximum Principle	104
4.4.2	Frechet Derivative of the Fit Error	105
4.4.3	Reduced Dynamics and Jump Discontinuities on $\mathfrak{so}^*(3)$	107
4.4.4	Explicit Solution of the Reduced Dynamics	109
4.5	Example II: Data Smoothing on $SE(2)$	111
4.5.1	Maximum Principle	114
4.5.2	Frechet Derivative of the Fit Error	115
4.5.3	Reduced Dynamics and Jump Discontinuities on $\mathfrak{se}^*(2)$	117
4.5.4	Explicit Solution of the Reduced Dynamics	119
4.6	Conclusion	120

II	Analysis of Collectives	121
5	Analysis of Bat Foraging in Two Different Contexts	122
5.1	Experiment Details and Reconstruction of Trajectories	124
5.2	Pre-processing of Trajectory Data	127
5.3	Analysis of Flight Strategy	132
5.4	Analysis of Steering Control	137
5.5	Discussion	146
6	Analysis of Flocking in European Starlings	149
6.1	Experiment Details and Trajectory Reconstruction	151
6.2	Analysis of Flight Strategy and Underlying Steering Control	154
6.3	Discussion	160
III	Synthesis of Collectives	162
7	Flocking through Topological Velocity Alignment (TVA)	163
7.1	Topological Velocity Alignment (TVA) Strategy	164
7.2	TVA Strategy for a Planar 2-agent System	168
7.2.1	State Space and Its Reduction onto Shape Space	169
7.2.2	Shape Dynamics	171
7.2.3	Analysis of TVA Feedback Law	174
7.2.4	Extension to a Three Dimensional Setting	180
7.3	TVA Strategy for an N -agent System with Cyclic Interaction	181
7.3.1	State Space and Its Reduction onto Shape Space	182
7.3.2	Shape Dynamics	184
7.3.3	Analysis of TVA Feedback Law	189
7.4	Algorithm for an N -agent System in a Three-Dimensional Setting	194
7.5	Implementation on Mobile Robot Testbed	196

7.5.1	Experimental Setup	196
7.5.2	Experiment I	198
7.5.3	Experiment II	199
7.5.4	Experiment III	200
8	Conclusion and Future Works	203
8.1	Summary of Contributions and Future Directions of Research	203
8.1.1	Reconstruction of Collectives	203
8.1.2	Analysis of Collective Behavior in Nature	210
8.1.3	Synthesis of Collective Motion	213

List of Figures

1	Natural Frenet frame for a curve in 3-dimensional space	11
2	Pursuit strategies: CP & MC	15
3	Multi resolution approach to solve optimization problem	32
4	Inclusion of virtual points to take care of the missing data points . .	33
5	Reconstruction of a circular helix	36
6	Reconstruction of a curve on a sphere	72
7	Performance comparison: synthetic data	72
8	Reconstruction of a bat-insect trajectory pair	74
9	Performance comparison: bat-insect trajectory pair (κ)	75
10	Performance comparison: bat-insect trajectory pair (ν)	75
11	Lie-Poisson Reduction	101
12	Unicycle Dynamics	112
13	A typical example of trajectory reconstruction (bat-mantis)	125
14	This figure shows the distribution of optimal values of regularization parameter (λ^*) used to reconstruct trajectories for bat-bat interactions.	126
15	This figure shows the distribution of optimal values of regularization parameter (λ^*) used to reconstruct trajectories for bat-mantis inter- actions.	127

16	This figure shows the CDF for contiguous durations of following <i>and</i> converging (Class <i>I</i>) flight segments.	131
17	Segmentation of flight behavior for the bat-bat interaction	131
18	Segmentation of flight behavior for the bat-mantis interaction	132
19	Strategy comparison for bat-bat interaction	133
20	Strategy comparison for bat-bat interaction (class I)	134
21	CDF comparison for bat-bat interaction	134
22	Strategy comparison for bat-mantis interaction	135
23	Strategy comparison for bat-mantis interaction (class I)	135
24	CDF comparison for bat-mantis interaction	136
25	Comparison of CATD and CP feedback laws for bat-bat interaction .	141
26	Residual analysis of CATD feedback law for bat-bat interaction . . .	142
27	Residual analysis of CP feedback law for bat-bat interaction	142
28	Comparison of CATD and CP feedback laws for bat-mantis interaction	143
29	Residual analysis of CATD feedback law for bat-mantis interaction .	144
30	Residual analysis of CP feedback law for bat-mantis interaction . . .	144
31	Cross validation (OCV) results for flocking events	152
32	Reconstruction of flocking events	153
33	Variation of flock averaged coherence w.r.t. neighborhood size	157
34	Variation in correlation of curvatures w.r.t. neighborhood size and behavioral latency	158
35	Individual trajectories along with their frame vectors	165
36	Variation of Θ_i with change in angle between \mathbf{x}_i and \mathbf{x}_{N_i}	167
37	Illustration of scalar shape variables for a 2-agent system	171
38	Illustration of TVA strategy for a 2 agent system	176
39	Phase portraits for the restricted dynamics	179

40	Illustration of an N -agent cyclic interaction system	182
41	Illustration of scalar shape variables for an N -agent cyclic interaction system	185
42	Pioneer 3-DX Mobile Robot	197
43	Experimental Setup at ISL, University of Maryland, College Park . .	198
44	Experiment I: 8 agents, Flocking	199
45	Experiment II: 8 agents, Splitting	200
46	Experiment III: 6 agents, Perturbation	201

List of Tables

2.1	Variation of OCV estimate of λ with different values of signal-to-noise ratio (SNR).	35
5.1	Theoretically plausible feedback laws for constant absolute target direction (CATD/MC) and classical pursuit (CP).	138
5.2	Summary of the statistical analysis of steering control laws for bat-bat pursuit events	143
5.3	Summary of the residual analysis of steering control laws for bat-bat pursuit events	143
5.4	Summary of the statistical analysis of steering control laws for bat-mantis pursuit events	146
5.5	Summary of the residual analysis of steering control laws for bat-mantis pursuit events	146
6.1	Summary of Flocking Events	154
6.2	Summary of the statistical analysis of steering control laws for starling flocks	159

List of Abbreviations

CATD	Constant Absolute Target Direction
CP	Classical Pursuit
DMI	Directed Mutual Information
MC	Motion Camouflage
OCV	Ordinary Cross Validation
ODE	Ordinary Differential Equation
PMP	Pontryagin's Maximum Principle
ROS	Robot Operating System
SNR	Signal to Noise Ratio
TVA	Topological Velocity Alignment
UAV	Unmanned Air Vehicle

Notations

$SE(n)$	Special Euclidean group of dimension n
$SO(n)$	Special orthogonal group of dimension n
S^n	n -dimensional sphere
$se(n)$	Lie algebra of $SE(n)$
$so(n)$	Lie algebra of $SO(n)$
\mathbb{I}_n	Identity matrix of size $n \times n$
$SO(3)$	Lie-group of 3×3 orthogonal Matrices
$so(3)$	Vector space of 3×3 skew-symmetric matrices
$O(\epsilon)$	Big O -notation (Infinitesimal asymptotics)

Chapter 1: Introduction

1.1 Background and Motivation

Collective motion plays a pivotal role in modern robotics and engineering, especially in the area of search and rescue missions (Andriluka *et al.* [2010]; Liu & Nejat [2013]; Marques *et al.* [2006]), surveillance (Ahmadzadeh *et al.* [2006]; Bethke *et al.* [2009]; Harmon [1987]; Kingston *et al.* [2008]; Li *et al.* [2008]; Quigley *et al.* [2005]), environmental monitoring (Dunbabin & Marques [2012]; Elfes *et al.* [1998]; Leonard *et al.* [2010]; Pinto *et al.* [2013]; Tokekar *et al.* [2010, 2013]) etc. On the other hand, examples of collective motion can be observed in a variety of natural settings. Fish schools [Gautrais *et al.*, 2009], locust migratory bands [Bazazi *et al.*, 2008], pigeon flights [Nagy *et al.*, 2010], starling murmurations [Ballerini *et al.*, 2008b] - examples are ubiquitous in nature. The reasons for *living in a group* are relatively well studied, and researchers have attributed a range of evolutionary factors behind group motion, including better defense against predator attack [Carere *et al.*, 2009], energy efficient movement due to aerodynamic (Cutts & Speakman [1994]; Weimerskirch *et al.* [2001]) or hydrodynamic [Herskin & Steffensen, 1998] coupling, information sharing [Miller *et al.*, 2013], cooperative food collection [Beekman *et al.*, 2001], and others. However, the individual-level interaction mechanisms, that give

rise to group level collective behavior, are not yet well-understood. Therefore, it seems to be a relevant effort to explore the underlying strategies and control laws governing collective motion, because it is not only beneficial from the perspective of engineering adaptation and exploitation, but it will also further our basic scientific understanding of nature. Also, with recent advances in bio-inspired designs for unmanned vehicles (Roberts *et al.* [2014]; Sfakiotakis *et al.* [2014]; Tan *et al.* [2006]), the importance of understanding the interaction mechanism (governing collective behavior) is becoming more and more prominent. The primary objective of this dissertation is to contribute to both the analysis of collective motion arising in natural settings, and the synthesis of biomimetic, decentralized algorithms for collective motion.

As the first step towards our goal, we identify the need to recover a trajectory, along with its higher order derivatives, from a finite set of discrete, probably noisy, observations. Generally, in the field of neuroethology and bio-inspired robotics, the research is often aimed towards exploration of the underlying strategies and steering control laws governing pursuit (Chiu *et al.* [2010]; Olberg *et al.* [2000]) and collective motion (Ballerini *et al.* [2008b]; Nagy *et al.* [2010]), and an integral part of this study involves analysis of relevant parameters of motion (namely curvature, speed, lateral acceleration etc.). Although the existing techniques for trajectory reconstruction use cubic splines or smoothing filters, in many cases they lack appropriate physical justification. From a broader perspective, the problem of recovering a smoothed signal from noisy observations appears in many areas of science and engineering, and belongs to a broader class of ill-posed (due to non-uniqueness, high sensitivity

to noise) inverse problems.

In our approach, we tackle the lack of well-posedness of this inverse problem by the method of regularization (Tikhonov [1963]; Wahba [1990]), and embed this problem into a proper hypothesis space. With an intention to leverage the techniques from optimal control theory, we introduce generative models (governed by differential equations) with inputs, states and outputs, and construct the hypothesis space as output of this generative model (given an input). This framework allows us to turn data smoothing into a continuous time optimal control problem with intermediate state costs, and the choice for path cost specifies the regularization imposed on the data.

Some initial works along a similar line have previously been carried out by Magnus Egerstedt, Clyde Martin and their collaborators. Zhang *et al.* [1997] have introduced the idea that spline functions can be constructed using techniques from linear control theory, and developed polynomial interpolating splines using linear generative models. Later, by adopting a variational approach for linear time invariant generative models, Shan *et al.* [2000] develops the necessary framework to recover a smoothing input (scalar) from noisy samples of output data (scalar). However, this analysis does not provide much insight about extending the results for a linear multi input multi output system. Later works (Kano *et al.* [2008]; Zhou *et al.* [2005, 2006]) along this line address some related aspects, and a book by Egerstedt & Martin [2010] provides a nice survey of the key developments along this line.

Alternatively one can describe the trajectory using a nonlinear generative model (probably evolving on a Lie group), and by choosing appropriate control

inputs, this problem can be cast as a data smoothing problem in a sub-Riemannian setting. Over the last two decades, Peter Crouch, Fatima Silva Leite and their collaborators, by adopting a variational approach, have been exploring the problem of data smoothing in Riemannian contexts wherein the number of control inputs is same as the dimension of the underlying state space (Crouch & Leite [1991, 1995]; Jakubiak *et al.* [2006]; Machado *et al.* [2010]). In a more recent work, Brody *et al.* [2012] and Burnett *et al.* [2013], while studying the data smoothing problem in a quantum state transfer context, have used higher order Lagrangian (involving higher order derivatives of control inputs) to impose smoothness of the reconstructed trajectory.

As recently highlighted in the work of Ben-Yosef & Ben-Shahar [2012], this regularization based approach towards data smoothing can be used to address problems in computer vision. Drawing inspiration from neuro-physiological aspects of the primary visual cortex (V1), in particular the ice-cube model [Hubel, 1995], this recent work suggests that a curve completion problem can be formulated as a data smoothing problem on $SE(2)$.

Equipped with an appropriate algorithm for trajectory reconstruction and parameter extraction, we set our next goal as analyzing flight trajectory data and exploring feedback laws underlying pursuit and collective motion in natural settings. Although the history of studying collective motion in nature dates back to the first half of the last century [Spooner, 1931], studying collective behavior from a reverse engineering perspective did not start until very recently. Complete understanding of the sensory perception and motor control governing collective motion (or even

pursuit) still remains an open problem. Individual members of the group are believed to use simple and plausible rules (control laws) which can be represented in a coordinate free manner, or in other words the laws should depend only on the relative motion information. One of the primary aims of this dissertation is to study the underlying mechanism of starling flocks. Although researchers have studied different aspects of flocking behavior in European starlings Ballerini *et al.* [2008*a,b*]; Cavagna *et al.* [2010]; Young *et al.* [2013], the interaction laws are not yet understood. By modelling the individual members as self driven particles under gyroscopic control, we aim to gather some insight for this relatively unexplored aspect.

The existing models for collective motion can be classified into two distinct categories: (i) Eulerian or continuum; and (ii) Lagrangian or individualistic. The Eulerian model uses a set of partial differential equations to describe the spatio-temporal evolution of the group density of a collective. Although this modeling approach does not appear to be very relevant in the context of robotic implementations, it has shown its effectiveness in analyzing densely packed collectives (Kudrolli *et al.* [2008]; Topaz *et al.* [2006]; Zhang *et al.* [2010]).

On the other hand, in a more individualistic Lagrangian approach the dynamics of each member is influenced by the state of the other members of group (this survey paper by Vicsek & Zafeiris [2012] provides a comprehensive review of the research along these lines). Although the first algorithm for collective motion was developed by Aoki [1982] for simulating the motion of a fish school, it is not the most well-known paper in the field. *Boids algorithm*, proposed in a later work by Reynolds [1987], is more famous in the context of collective motion. However, both

of these approaches consider three different modes of interaction between the individuals, namely, (i) velocity alignment with the neighbors; (ii) approach towards the flock center of mass; and (iii) collision avoidance. It was also noted that collective behavior emerges in a leaderless manner.

Almost a decade later, Vicsek *et al.* [1995] introduced the notion of self driven particles for collective motion. By assuming equal and constant speed for individual agents, this work demonstrates the emergence of ordered motion in a planar setting. This flocking behavior is achieved by updating an agent's direction (at each time step) by the average direction of motion of its neighbors (i.e., agents within a fixed distance from the focal agent). Later, Jadbabaie *et al.* [2003] gave the much needed theoretical explanation for this flocking behavior. However, a later note by Bertsekas & Tsitsiklis [2007] has shown that the main convergence results of this work can be perceived as a special case of some earlier results by Tsitsiklis *et al.* [1986]. A more recent work by Cucker & Smale [2007] describes the evolution of a flock by considering a polynomial decay of the influence between individuals in the flock as they separate in space. A common theme in this line of works is that individual agents are assumed to be governed by first-order dynamics, which allows us to interpret the consensus dynamics as a discrete analogue of heat equation.

However, a recent work by Attanasi *et al.* [2014] suggests that directional information within a flock propagates with an almost constant speed, which is much faster than the diffusive transport of information predicted by first-order models with heat-like aspects. This observation provides sufficient justification to look beyond first-order consensus algorithms. Interestingly, over the last few years a shift of

attention towards second-order consensus algorithms has been noticed in the control community too (Olfati-Saber [2006]; Ren [2008]; Ren & Atkins [2005]; Yu *et al.* [2010, 2011]). In this dissertation we propose a strategy for collective motion which can explain how local interactions between neighbors can influence the agents to align their headings in a single common direction. Moreover, this strategy, which we refer as *topological velocity alignment* (TVA), has wave-like aspects and can explain fast propagation of information across the flock.

1.2 Thesis Outline

In Chapter 2, we define the problem of trajectory reconstruction as a nonlinear optimization problem and approach it from a mathematical programming perspective. By assuming the natural Frenet frame equations as the underlying generative model, and penalizing high rates of change in curvatures and speed, we turn the trajectory reconstruction problem into a regularized inverse problem. This regularized inverse problem can be viewed as an optimization over an infinite dimensional function space. We introduce mathematical reparametrization (via Caley transform, exponential mapping etc.) of the underlying variables to make the problem tractable, and then use a numerical routine (*fminunc* [fminunc]) to solve the optimization. Also, by noting the importance of cross validation in this context, we perform ordinary cross validation to compute an optimal amount of regularization.

As the previous approach was computationally very demanding, and doesn't guarantee minimization in a global sense, we develop an alternative linear formu-

lation of the trajectory reconstruction problem in Chapter 3. By using a triple integrator as the underlying generative model for trajectory evolution, we regularize the problem by trading total fit-error against high values of the jerk path integral. Then, by using techniques from linear quadratic optimal control theory, this approach yields an analytic solution for the true minimum of this problem.

In Chapter 4, we develop a framework to address data smoothing problems arising in a nonlinear setting. Then, by using a modified version of Pontryagin's maximum principle, we derive the necessary conditions for a solution of the regularized inversion problem. Moreover, as the natural Frenet frame equations for a trajectory can be viewed as a left-invariant dynamics on a Lie group, we extend this result to address data smoothing in a matrix Lie group setting. We end this chapter by discussing two example problem on (on $SO(3)$ and $SE(2)$, respectively).

Then we delve into the analysis of natural occurrences of collective motion. In Chapter 5, we consider the most basic form of collective motion, namely a dyadic interaction between two individuals (conspecific and conspecific). Here, we analyze the flight trajectory data provided by our collaborators from Auditory Neuroethology Laboratory, Department of Psychology, University of Maryland, and show evidences in favor of context specific switch in bat flight strategy. Our analysis also provides an estimate of the sensorimotor delay, associated with the pursuit behavior.

Chapter 6 describes the flight strategy analysis of flocking behavior in European starlings. Here, we begin by extracting speed and curvatures from the sampled dataset of observed positions. Then we perform correlation analysis to investigate the feedback mechanism for steering control governing coordinate motion of the

flocks.

Finally, drawing inspiration from our findings in the previous chapter, we propose a flight strategy, called topological velocity alignment (TVA), along with a plausible control law in Chapter 7. This strategy can be perceived as a restriction in the underlying state space, wherein each member of the flock aligns its velocity along the velocity of its neighborhood of center of mass. We complement our theoretical analysis of the proposed control law (in some special cases) by providing some implementation results.

1.3 Mathematical Preliminaries

Now we briefly describe some mathematical concepts which we will use throughout this thesis.

1.3.1 Self-steering Particle Model

The trajectory of a single particle moving in three dimensional space can be described by a function $\mathbf{r} : [0, T_{final}] \rightarrow \mathbb{R}^3$, where $T_{final} > 0$. If $t = t(s)$ is a smooth, real valued function and $\mathbf{r}(t)$ is a curve in \mathbb{R}^3 , we call the curve $\beta(s) \triangleq \mathbf{r}(t(s))$ a reparametrization of the curve \mathbf{r} . We assume $\mathbf{r}(t)$ to be a regular curve, i.e. $\dot{\mathbf{r}}(t) \neq 0 \quad \forall t \in [0, T_{final}]$. Now we define the arc length parameter as $s(t) = \int_0^t \|\dot{\mathbf{r}}(\sigma)\| d\sigma$. Under the regularity assumption, $s(t)$ is a continuous, strict monotone increasing function and hence invertible. Therefore a reparametrization of a curve can be obtained by using the arc length as the parameter and this partic-

ular parametrization has a special feature. Any curve parametrized using arc length parameter will have unit speed.

Let $\mathbf{r}_{mod}(s)$ be the reparametrization of $\mathbf{r}(t)$ using arc length parameter. The evolution of $\mathbf{r}_{mod}(s)$ in three dimensional space can be described using the *Frenet-Serret* framing of this curve, as this is standard in differential geometry [do Carmo, 1976]. But this approach requires the curve to be at least thrice continuously differentiable and we need the curvature ($\kappa(s) = \| \mathbf{r}_{mod}''(s) \|$) to be positive to avoid degeneracy in defining the normal direction. The requirement of non-zero curvature everywhere poses serious difficulties in this particular problem of our consideration (as the trajectories, to be reconstructed, may have point of inflection) and hence the Frenet-Serret framing is not the best choice for our purpose. So we use an alternate framing of the curve, the *Natural Frenet frame*, which is also known as the *Relatively Parallel Adapted Frame* [Bishop, 1975]. The natural Frenet frame is an alternative approach to define a moving frame that is well-defined even when the curve has vanishing second derivative.

The natural Frenet frame is based on the observation that, while the unit tangent vector $\mathbf{x}(t)$ for a given curve is unique, we can choose two unit vectors ($\mathbf{y}(t), \mathbf{z}(t)$) on the plane perpendicular to $\mathbf{x}(t)$ such that $\{\mathbf{x}(t), \mathbf{y}(t), \mathbf{z}(t)\}$ defines a right-handed orthogonal frame. The evolution of the frame along the length of the

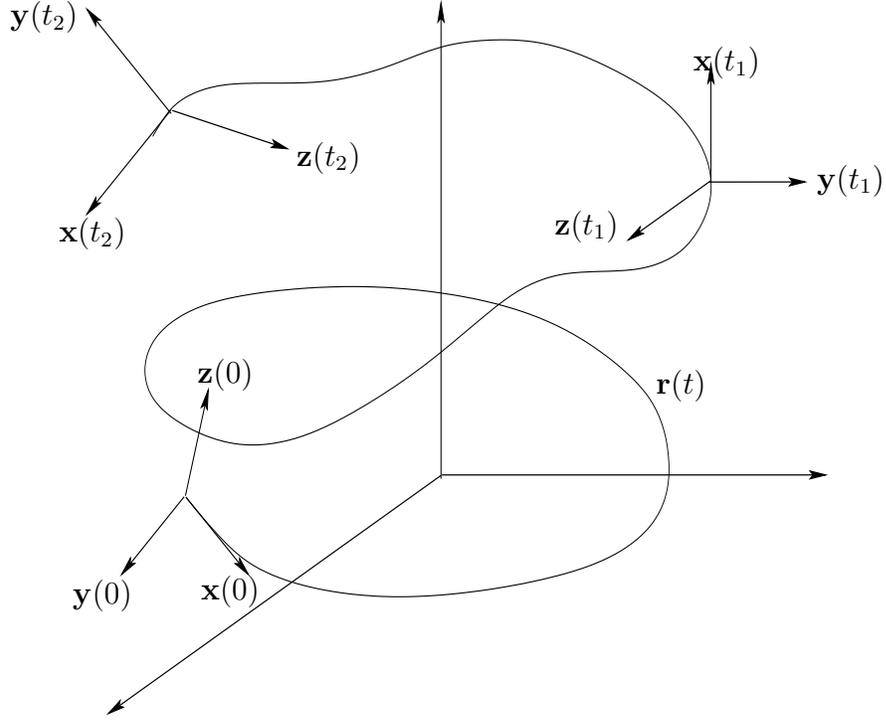


Figure 1: Natural Frenet frame for a curve in 3-dimensional space.

trajectory is governed by

$$\begin{aligned}
 \dot{\mathbf{r}}(t) &= \nu(t)\mathbf{x}(t) \\
 \dot{\mathbf{x}}(t) &= \nu(t)(u(t)\mathbf{y}(t) + v(t)\mathbf{z}(t)) \\
 \dot{\mathbf{y}}(t) &= -\nu(t)u(t)\mathbf{x}(t) \\
 \dot{\mathbf{z}}(t) &= -\nu(t)v(t)\mathbf{x}(t),
 \end{aligned} \tag{1.1}$$

where ν is the speed and (u, v) are the natural curvatures of the trajectory. In this approach for framing a regular curve, we can choose the initial orientation of the frame at our will as $\mathbf{y}(0)$ and $\mathbf{z}(0)$ can be chosen arbitrarily, in contrary to the Frenet-Serret frame equations where all three frame vectors are uniquely defined

once the curve is given (under a hypothesis of positive curvature). From $u(t)$ and $v(t)$ (which can be viewed as Cartesian components of the curvature) we can obtain the curvature $\kappa(t)$ and the torsion $\tau(t)$ using the following set of relations:

$$\kappa(t) = \sqrt{u^2(t) + v^2(t)}, \quad \theta(t) = \arctan\left(\frac{v(t)}{u(t)}\right), \quad \tau(t) = \frac{d\theta(t)}{dt}.$$

1.3.2 Pursuit Strategies and Feedback Laws

1.3.2.1 Classical pursuit

In classical pursuit (CP) the follower (pursuer) moves directly towards the evader (target). By representing the evader position by \mathbf{r}_e , pursuer position by \mathbf{r}_p , and pursuer direction of motion by \mathbf{x}_p , we can define a contrast function (Λ) as

$$\Lambda = \mathbf{x}_p \cdot \frac{\mathbf{r}}{|\mathbf{r}|}, \tag{1.2}$$

to measure how closely the leader-follower relationship matches the CP strategy, where $\mathbf{r} = \mathbf{r}_p - \mathbf{r}_e$. This contrast function can be interpreted as the cosine of the angle between the baseline vector and the velocity of the pursuer and $\Lambda = -1$ implies that the pursuer is on the CP manifold.

Proposition 1.1 (Galloway *et al.* [2010], Proposition 1). *Let us assume that the dynamics of the pursuer (\mathbf{r}_p) and the evader (\mathbf{r}_e) trajectories be governed by the natural Frenet frame equations, with (ν_p, u_p, v_p) and (ν_e, u_e, v_e) being the speed and*

curvatures, respectively. Consider the feedback control law given by

$$\begin{aligned} u_p &= \frac{1}{\nu_p} \left[-\mu \left(\mathbf{y}_p \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right) - \frac{1}{|\mathbf{r}|} \left(\mathbf{z}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{|\mathbf{r}|} \right) \right) \right] \\ v_p &= \frac{1}{\nu_p} \left[-\mu \left(\mathbf{z}_p \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right) + \frac{1}{|\mathbf{r}|} \left(\mathbf{y}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{|\mathbf{r}|} \right) \right) \right] \end{aligned} \quad (1.3)$$

where $\mu > 0$ is the feedback gain. Then $\Lambda(t) \rightarrow -1$ as $t \rightarrow \infty$, whenever $\Lambda(0) \notin \{1, -1\}$.

Proof. We begin by differentiating Λ along the trajectories of the generative model for both pursuer and evader.

$$\begin{aligned} \dot{\Lambda} &= \dot{\mathbf{x}}_p \cdot \frac{\mathbf{r}}{|\mathbf{r}|} + \mathbf{x}_p \cdot \frac{d}{dt} \left(\frac{\mathbf{r}}{|\mathbf{r}|} \right) \\ &= \nu_p (u_p \mathbf{y}_p + v_p \mathbf{z}_p) \cdot \frac{\mathbf{r}}{|\mathbf{r}|} + \frac{1}{|\mathbf{r}|} \mathbf{x}_p \cdot \left(\dot{\mathbf{r}} - \left(\dot{\mathbf{r}} \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right) \frac{\mathbf{r}}{|\mathbf{r}|} \right). \end{aligned} \quad (1.4)$$

Now we define the transverse component of the relative velocity as

$$\mathbf{w} = \dot{\mathbf{r}} - \left(\dot{\mathbf{r}} \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right) \frac{\mathbf{r}}{|\mathbf{r}|},$$

and by using the BAC-CAB identity, $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) = \mathbf{b} \cdot (\mathbf{a} \times \mathbf{c}) - \mathbf{c} \cdot (\mathbf{a} \times \mathbf{b})$, for arbitrary vectors \mathbf{a} , \mathbf{b} and \mathbf{c} , we note that

$$\begin{aligned} \mathbf{w} &= \dot{\mathbf{r}} \left(\frac{\mathbf{r}}{|\mathbf{r}|} \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right) - \frac{\mathbf{r}}{|\mathbf{r}|} \left(\frac{\mathbf{r}}{|\mathbf{r}|} \cdot \dot{\mathbf{r}} \right) \\ &= \frac{\mathbf{r}}{|\mathbf{r}|} \times \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{|\mathbf{r}|} \right). \end{aligned} \quad (1.5)$$

As the feedback law is governed by (1.3) and the transverse component of the relative velocity can be expressed as a vector triple product (1.5), we can re-write (1.4) as

$$\begin{aligned}
\dot{\Lambda} &= -\mu \left[\left(\mathbf{y}_p \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right)^2 + \left(\mathbf{z}_p \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right)^2 \right] + \frac{1}{|\mathbf{r}|} \mathbf{x}_p \cdot \mathbf{w} \\
&\quad - \frac{1}{|\mathbf{r}|} \left[\left(\mathbf{z}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{|\mathbf{r}|} \right) \right) \left(\mathbf{y}_p \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right) - \left(\mathbf{y}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{|\mathbf{r}|} \right) \right) \left(\mathbf{z}_p \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right) \right] \\
&= -\mu \left[\left(\frac{\mathbf{r}}{|\mathbf{r}|} \cdot \frac{\mathbf{r}}{|\mathbf{r}|} \right)^2 - \left(\frac{\mathbf{r}}{|\mathbf{r}|} \cdot \mathbf{x}_p \right)^2 \right] + \frac{1}{|\mathbf{r}|} \mathbf{x}_p \cdot \mathbf{w} \\
&\quad - \frac{1}{|\mathbf{r}|} (\mathbf{z}_p \times \mathbf{y}_p) \cdot \left(\left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{|\mathbf{r}|} \right) \times \frac{\mathbf{r}}{|\mathbf{r}|} \right) \\
&= -\mu (1 - \Lambda^2) + \frac{1}{|\mathbf{r}|} [\mathbf{x}_p \cdot \mathbf{w} - (\mathbf{y}_p \times \mathbf{z}_p) \cdot \mathbf{w}]. \tag{1.6}
\end{aligned}$$

Now $\mathbf{y}_p \times \mathbf{z}_p = \mathbf{x}_p$ because $\{\mathbf{x}_p, \mathbf{y}_p, \mathbf{z}_p\}$ forms an orthonormal triad. Therefore, we have

$$\dot{\Lambda} = -\mu (1 - \Lambda^2), \tag{1.7}$$

and it is clear from (1.7) that whenever $\Lambda \in (-1, 1)$ it results in $\dot{\Lambda} < 0$. In fact it can be concluded that the level sets of $\{\Lambda = 1\}$ and $\{\Lambda = -1\}$ are two invariant manifolds under the closed loop dynamics.

Moreover, by assuming $\Lambda(0) \neq \pm 1$, we have

$$\Lambda(t) = \frac{K e^{-2\mu t} + 1}{K e^{-2\mu t} - 1} \tag{1.8}$$

where the constant K is defined as $K = \frac{\Lambda_0 + 1}{\Lambda_0 - 1}$. Since $e^{-2\mu t} \rightarrow 0$ as $t \rightarrow \infty$, we have $\Lambda(t) \rightarrow -1$ as $t \rightarrow \infty$. \square

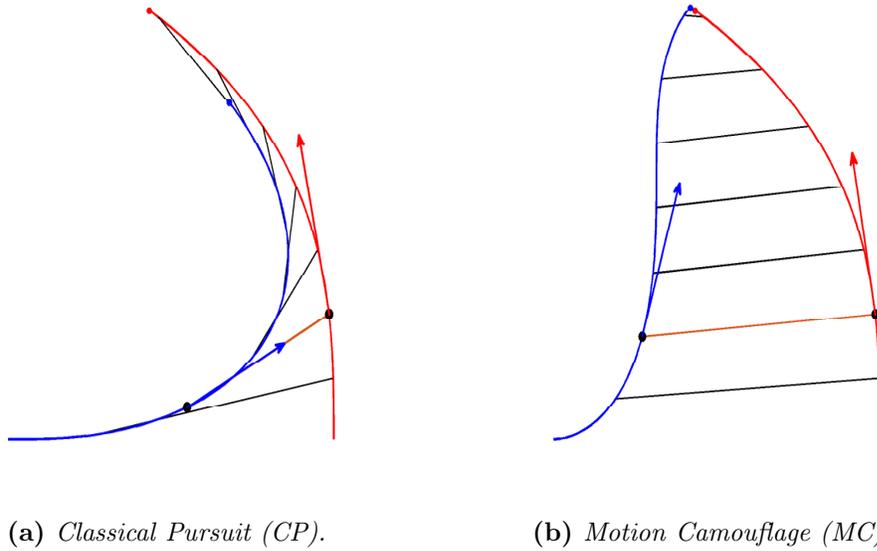


Figure 2: These figures illustrate the strategies to be examined, namely (a) *Classical Pursuit* and (b) *Motion Camouflage*. The red curve and arrow represents the evader trajectory, along with its velocity. In a similar way, the blue curve and arrow represents the pursuer trajectory, along with its velocity. The black lines corresponds to the baselines connecting the pursuer with the evader.

1.3.2.2 Motion camouflage

In motion camouflage (MC) (with respect to infinity) the follower approaches the leader in such a way that the relative velocity does not have any transverse component with respect to the baseline vector [Justh & Krishnaprasad, 2006]. Motion camouflage with respect to infinity is a stealthy pursuit because it nullifies the motion parallax. Similar to CP one can define a contrast function to measure how closely the leader-follower relationship matches the MC strategy. One such contrast function is given by

$$\Gamma = \frac{\mathbf{r}}{|\mathbf{r}|} \cdot \frac{\dot{\mathbf{r}}}{|\dot{\mathbf{r}}|}, \quad (1.9)$$

where the baseline vector is defined as $\mathbf{r} = \mathbf{r}_p - \mathbf{r}_e$. One can easily check that Γ is the cosine of the angle between the baseline vector and the relative velocity

of the follower with respect to the leader. Moreover $\Gamma = -1$ implies that the follower is on the MC manifold. MC is also referred as the constant absolute target direction strategy (CATD) because the direction of the baseline vector remains fixed throughout the pursuit. Now we recall the following result by Reddy *et al.* [2006] to show finite time accessibility of motion camouflage in three dimensions.

Proposition 1.2 (Reddy *et al.* [2006]). *Let us assume that the dynamics of the pursuer (\mathbf{r}_p) and the evader (\mathbf{r}_e) trajectories are governed by the natural Frenet frame equations, with (ν_p, u_p, v_p) and (ν_e, u_e, v_e) being the speed and curvatures, respectively. Consider the feedback control law given by*

$$\begin{aligned} u_p &= -\mu \left(\mathbf{z}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{|\mathbf{r}|} \right) \right) \\ v_p &= \mu \left(\mathbf{y}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{|\mathbf{r}|} \right) \right). \end{aligned} \tag{1.10}$$

Moreover we assume the following hypotheses to be true:

(A1) $0 < \nu_p^{low} \leq \nu_p \leq \nu_p^{high} < \infty$, where ν_p^{low} and ν_p^{high} are constants.

(A2) $0 < \nu_e^{low} \leq \nu_e \leq \nu_e^{high} < \infty$, where ν_e^{low} and ν_e^{high} are constants.

(A3) $\nu_e/\nu_p \leq \nu_{MAX} < 1$, where ν_{MAX} is a constant.

(A4) u_e and v_e are piecewise continuous, and $\sqrt{u_e^2 + v_e^2}$ is bounded.

(A5) $\dot{\nu}_e$ and $\dot{\nu}_p$ are piecewise continuous; $|\dot{\nu}_e| < \alpha_e$ and $|\dot{\nu}_p| < \alpha_p$ where α_e and α_p are finite constants.

(A6) $\Gamma(0) < 1$ and $|\mathbf{r}(0)| > 0$.

Then motion camouflage is accessible in finite time using high-gain feedback, i.e. by choosing $\mu > 0$ to be sufficiently large.

In a later work Reddy *et al.* [2007] have shown that motion camouflage is accessible in finite time even when some amount of delay is incorporated into the feedback law (under some constraints on the feedback gain, delay and relative speed).

Part I

Reconstruction of Collectives

Chapter 2: Data Smoothing through Nonlinear Optimization - Mathematical Programming

The problem of recovering a smoothed signal from noisy observations appears in many areas of science and engineering, and belongs to a broader class of *inverse problems*. By noting that naive solutions are non-unique and highly sensitive to noise, it can be concluded that this inverse problem is ill-posed. However, this lack of well-posedness can be tackled through a regularized approach (Tikhonov [1963]; Wahba [1990]). The main idea behind regularization is to embed the problem of interest into a hypothesis space and minimize a cost functional expressed as a sum of two terms: (i) misfit of a hypothesis to observed data; and (ii) a penalty functional accounting for complexity of a hypothesis. In our context, we build the hypothesis space by introducing generative models governed by ordinary differential equations with inputs, states and outputs. This yields the hypothesis as output of the generative model, given an input (control). This set-up allows us to turn data smoothing into a continuous time optimal control problem with intermediate state costs.

Regularization, in our context, necessitates penalizing sharp turns in the trajectory. By assuming an appropriate generative model for trajectory evolution, this

problem of trajectory reconstruction can be formulated as an optimization problem, wherein the regularization term is treated as the Lagrangian, and the fit-errors (between observed data and smoothed data obtained from the generative model) constitute the intermediate and terminal costs. This problem becomes quite relevant when we attempt to analyze the strategies and feedback mechanisms governing biological collectives (starling flocks, foraging bats) because the analysis requires studying parameters of motion (speed, curvatures *etc.*) and this information is available only after the inverse problem has been solved.

An important aspect of this approach is to estimate an optimal value for the relative weight of the regularization term with respect to the sum of fit-errors, and then use this estimated value to reconstruct the smoothed signal by solving the optimization problem. In our study, relative weight of the regularization term has been represented by introducing a smoothing parameter (λ) into the optimization cost. Clearly, the choice of this smoothing parameter is critical to the nature of a reconstructed trajectory. It controls the balance between the goodness of fit and the smoothness of the regression function. The optimization algorithm will produce a very wiggly estimate for low values of the smoothing parameter, and at large values the goodness of fit will deteriorate. So a trade-off is required to choose an appropriate value for the smoothing parameter. As the value of λ varies from 0 to ∞ , the estimate transforms itself from an interpolant of given data points to a geodesic best fitting the data points in a least square manner.

The estimation of the smoothing parameter has been done using *ordinary cross validation* technique. The main idea behind the cross validation technique

is the partitioning of given dataset into two subsets, namely the estimation subset and the validation subset. Once the signal is reconstructed using data from the estimation subset, the validation subset is used to examine the performance of the reconstruction. Cross validation attempts to minimize error between the predicted signal and the original signal over the validation data.

The inherent nonlinearity of the underlying generative model results in the following interesting features:

- System nonlinearity makes it very difficult to obtain a closed form solution for the transition matrix of the underlying generative model, and hence we need to use *geometric integration methods*.
- Lack of closed form solution for the cross validation cost function forces us to compute the cost over a finite set of smoothing parameter values.
- The cross validation technique has an inherent parallel structure which can be exploited to accelerate the computational process.

Although the main ideas that we pursue in this chapter are similar to the ones developed by Reddy [2007], they differ in the following aspects:

- In contrast to the previous approach we have used the initial position ($\mathbf{r}(0)$) as an optimizing variable. Earlier work assumed the initial position to be same as the first data point (r_0).
- We have used different ways to parametrize the rotation matrix for initial frame orientation (Cayley transform, instead of Euler angles) and the speed of the trajectory (using an exponential function we have shown bijection between \mathbb{R} and \mathbb{R}^+).

- The most distinctive difference lies in the way we choose a smoothing parameter (i.e. λ in (2.1)). In comparison to previously adopted heuristic approach, we implemented an algorithmic way (ordinary cross validation) to select an optimal value for the smoothing parameter.

2.1 Regularized Inversion Problem

Treating the feature point as a self-steering particle in three dimensions, a natural generative model for its position $\mathbf{r}(\cdot)$ is given by the natural Frenet frame equations (1.1), and the existence and uniqueness of this generative model has previously been discussed in the work of Bishop [1975]. Now, drawing inspiration from findings in biomechanics, we impose regularization by trading total fit-error against high rates of change in speed and curvatures. Hence, by letting $\{r_i\}_{i=0}^N$ denote the set of observed positions, one can formulate trajectory reconstruction as the following optimal control problem

$$\begin{aligned} & \underset{\substack{[\mathbf{x}, \mathbf{y}, \mathbf{z}](t_0), \\ \mathbf{r}(t_0), u, v, \nu}}{\text{Minimize}}}{\left(\sum_{i=0}^N \|\mathbf{r}(t_i) - r_i\|^2 + \lambda \int_{t_0}^{t_N} (\dot{u}^2 + \dot{v}^2 + \dot{\nu}^2) dt \right)} \\ & \text{subject to} \quad \text{System dynamics: Natural Frenet Frame (1.1),} \end{aligned} \tag{2.1}$$

$$[\mathbf{x} \ \mathbf{y} \ \mathbf{z}](t_0) \in SO(3), \quad \mathbf{r}(t_0) \in \mathbb{R}^3,$$

$$u : [t_0, t_N] \rightarrow \mathbb{R}, \quad v : [t_0, t_N] \rightarrow \mathbb{R}, \quad \nu : [t_0, t_N] \rightarrow \mathbb{R}^+,$$

where the smoothing parameter ($\lambda > 0$) is evaluated through the method of cross validation.

Alternatively, we can pack the position vector $\mathbf{r}(t)$, along with frame vectors $\mathbf{x}(t)$, $\mathbf{y}(t)$ and $\mathbf{z}(t)$, inside a 4×4 matrix $g(t)$ defined as

$$g(t) = \begin{bmatrix} R(t) & \mathbf{r}(t) \\ 0 & 1 \end{bmatrix} \in SE(3), \quad (2.2)$$

where $R(t) \triangleq [\mathbf{x}(t) \ \mathbf{y}(t) \ \mathbf{z}(t)] \in SO(3)$ represents the frame moving along the trajectory. Then, by letting $\xi_0 = E_{14}$; $\xi_2 = E_{13} - E_{31}$; $\xi_3 = E_{21} - E_{12}$ represent the standard basis elements of $se(3)$, and $\{e_i\}_{i=1}^4$ represent the set of standard basis vectors in \mathbb{R}^4 , the nonlinear generative model, *i.e.* the natural Frenet frame equations, of a curve (1.1) can be expressed as

$$\begin{aligned} \dot{g} &= g\xi \\ \mathbf{r} &= [e_1 \ e_2 \ e_3]^T g e_4, \end{aligned} \quad (2.3)$$

where ξ is given by

$$\xi = \nu(\xi_0 + u\xi_3 - v\xi_2) = \begin{bmatrix} 0 & -\nu u & -\nu v & \nu \\ \nu u & 0 & 0 & 0 \\ \nu v & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \quad (2.4)$$

Clearly, this generative model (2.3) can be perceived as a left-invariant dynamics on $SE(3)$, and hence the problem of trajectory reconstruction (2.1) can be treated as a data smoothing problem in a sub-Riemannian setting wherein the

number of controls (3) is strictly less than the state-space dimension (6). Although some initial results along this line has been proposed by Brody *et al.* [2012] and Dey & Krishnaprasad [2014a], further work is required before it leads us to a solution for this problem (2.1). On the other hand, we can treat (2.1) as an optimization problem with constraints (due to the nonlinear generative model) on $SE(3)$. Then, by expressing the trajectory $\mathbf{r}(\cdot)$ as a function of speed ($\nu(\cdot)$) and curvatures ($u(\cdot)$, $v(\cdot)$), it can be turned into an unconstrained optimization problem. The difficulties associated with obtaining a closed form solution of $g(\cdot)$ forces us to use special approximation techniques.

Remark 2.1. *It should be noted that λ , the smoothing parameter in the optimization problem (2.1), is not a unit-free quantity. Rather, it has a dimension of $[L^4T]$ where L and T represents the dimension of length and time, respectively. Also, to avoid a dimension mismatch inside the integrand of the cost function (2.1), we assume that a unit scaling factor of dimension $[L^{-4}T^2]$ has been applied to \dot{v}^2 .*

2.2 Some Practical Issues

2.2.1 Numerical Integration for the Group Dynamics

One can easily check that the solution for natural Frenet frame equation (2.3) can be expressed as

$$g(t) = g(0)\Phi^T(t, t_0), \quad t \in [t_0, t_N], \quad (2.5)$$

where the state transition matrix $\Phi(t, t_0)$ is computed as the limit point of the following Peano-Baker series

$$\Phi(t, t_0) = \mathbb{I}_4 + \int_{t_0}^t \xi^T(\sigma_1) d\sigma_1 + \int_{t_0}^t \xi^T(\sigma_1) \int_{t_0}^{\sigma_1} \xi^T(\sigma_2) d\sigma_2 d\sigma_1 + \cdots, \quad t \geq t_0. \quad (2.6)$$

However, finding a closed form solution for $\Phi(t, t_0)$ for any general $\xi(\cdot)$ is very challenging, and this difficulty forces us to adopt a numerical approach. As the dynamical constraint evolves over $SE(3)$, special care has to be taken while choosing a suitable numerical approach because otherwise numerical computation might force $g(t_j)$ to leave the manifold for some $j \in \{0, 1, 2, \dots, N\}$.

To address this issue we adopt a geometric integration method which ensures that the solution will lie on $SE(3)$ at every sample point. However, this advantage is gained at the cost of some freedom on the choice of speed and curvatures, in particular by assuming the speed and curvatures to be piecewise-constant functions. Now we consider a refined partition of the time interval $[t_0, t_N]$, given by $\{t_0 = t_{0_r} < t_{1_r} < t_{2_r} < \cdots < t_{K_r} = t_N\}$, where the length of these K_r number of equal intervals is given as $\delta = \frac{t_N - t_0}{K_r}$. By using the semi-group property of state transition matrices, we have

$$\Phi(t_{k_r}, t_0) = \Phi(t_{k_r}, t_{(k-1)_r}) \Phi(t_{(k-1)_r}, t_{(k-2)_r}) \cdots \Phi(t_{2_r}, t_{1_r}) \Phi(t_{1_r}, t_0),$$

and this provides a recursive representation of $g(t_{k_r})$, given by

$$g(t_{k_r}) = g(t_{(k-1)_r})\Phi^T(t_{k_r}, t_{(k-1)_r}). \quad (2.7)$$

Now, by exploiting the piecewise-constant nature of speed and curvatures, we assume $u(t) = u_k$, $v(t) = v_k$ and $\nu(t) = \nu_k$ over the interval $[t_{(k-1)_r}, t_{k_r}]$. These assumptions make $\xi(\cdot)$ constant on $[t_{(k-1)_r}, t_{k_r}]$, and as a consequence the state-transition matrix $\Phi^T(t_{k_r}, t_{(k-1)_r})$ can be represented as

$$\Phi(t_{k_r}, t_{(k-1)_r}) = \begin{bmatrix} \cos(\delta\alpha_k\nu_k) & \frac{u_k}{\alpha_k} \sin(\delta\alpha_k\nu_k) & \frac{v_k}{\alpha_k} \sin(\delta\alpha_k\nu_k) & 0 \\ -\frac{u_k}{\alpha_k} \sin(\delta\alpha_k\nu_k) & \frac{(v_k^2 + u_k^2 \cos(\delta\alpha_k\nu_k))}{\alpha_k^2} & \frac{u_k v_k (\cos(\delta\alpha_k\nu_k) - 1)}{\alpha_k^2} & 0 \\ -\frac{v_k}{\alpha_k} \sin(\delta\alpha_k\nu_k) & \frac{u_k v_k (\cos(\delta\alpha_k\nu_k) - 1)}{\alpha_k^2} & \frac{(u_k^2 + v_k^2 \cos(\delta\alpha_k\nu_k))}{\alpha_k^2} & 0 \\ \frac{1}{\alpha_k} \sin(\delta\alpha_k\nu_k) & \frac{u_k(1 - \cos(\delta\alpha_k\nu_k))}{\alpha_k^2} & \frac{v_k(1 - \cos(\delta\alpha_k\nu_k))}{\alpha_k^2} & 1 \end{bmatrix}, \quad (2.8)$$

where $\alpha_k = \sqrt{u_k^2 + v_k^2}$. Finally, by using the expressions from (2.2) and (2.8), (2.7) yields the following recursive equations to represent evolution of the trajectory $\mathbf{r}(\cdot)$, along with the natural Frenet frame $R(\cdot)$,

$$R(t_k) = R(t_{k-1}) \begin{bmatrix} \cos(\delta\alpha_k\nu_k) & -\frac{u_k}{\alpha_k} \sin(\delta\alpha_k\nu_k) & -\frac{v_k}{\alpha_k} \sin(\delta\alpha_k\nu_k) \\ \frac{u_k}{\alpha_k} \sin(\delta\alpha_k\nu_k) & \frac{(v_k^2 + u_k^2 \cos(\delta\alpha_k\nu_k))}{\alpha_k^2} & \frac{u_k v_k (\cos(\delta\alpha_k\nu_k) - 1)}{\alpha_k^2} \\ \frac{v_k}{\alpha_k} \sin(\delta\alpha_k\nu_k) & \frac{u_k v_k (\cos(\delta\alpha_k\nu_k) - 1)}{\alpha_k^2} & \frac{(u_k^2 + v_k^2 \cos(\delta\alpha_k\nu_k))}{\alpha_k^2} \end{bmatrix} \quad (2.9)$$

$$\mathbf{r}(t_k) = \mathbf{r}(t_{k-1}) + \frac{1}{\alpha_k} [\alpha_k \sin(\delta\alpha_k\nu_k)\mathbf{x}(t_{k-1}) + u_k(1 - \cos(\delta\alpha_k\nu_k))\mathbf{y}(t_{k-1}) + v_k(1 - \cos(\delta\alpha_k\nu_k))\mathbf{z}(t_{k-1})]. \quad (2.10)$$

Thus we have solved the differential equation and a closed form solution for the trajectory has been achieved under the piecewise constant assumption.

2.2.2 Parametrization of $SO(3)$ Using Cayley Transform

Moreover, we let initial orientation of the natural frame ($R(t_0)$) to be another variable for optimization. But $R(t_0)$ lies on a nonlinear manifold ($SO(3)$), and an optimization on a manifold is not as straightforward as an one on a Euclidean space \mathbb{R}^n . This forces us to adopt a suitable way to parametrize $SO(3)$. Although there are various methods for parametrizing $SO(3)$, we choose the approach using *Cayley transform*. There is another popular approach using the Euler angles but this approach suffers from the singularity issue which arises from the fact that $SO(3)$ and $S^1 \times S^1 \times S^1$ are not equivalent topologically.

Clearly, for any $\Theta \in SO(3)$ and any $x \in \mathbb{R}^3$, we have

$$\begin{aligned}
& \langle \Theta x, \Theta x \rangle = \langle x, \Theta^T \Theta x \rangle \\
\Rightarrow & \langle \Theta x, \Theta x \rangle = \langle x, x \rangle \text{ as } \Theta \in SO(3) \\
\Rightarrow & \langle \Theta x, \Theta x \rangle - \langle x, x \rangle = 0 \\
\Rightarrow & \langle (\Theta + \mathbb{I}_3)x, (\Theta - \mathbb{I}_3)x \rangle = 0. \tag{2.11}
\end{aligned}$$

Now we introduce a new variable z defined as $z = (\Theta + \mathbb{I}_3)x$. By assuming -1 not to be an eigen-value of Θ , we can show that $(\Theta + \mathbb{I}_3)$ is invertible and its column

space is the full space \mathbb{R}^3 . Then (2.11) can be equivalently represented as

$$\langle z, (\Theta - \mathbb{I}_3)(\Theta + \mathbb{I}_3)^{-1}z \rangle = 0, \quad \forall z \in \mathbb{R}^3. \quad (2.12)$$

It can easily be concluded from (2.12), that $(\Theta - \mathbb{I}_3)(\Theta + \mathbb{I}_3)^{-1}$ is a skew-symmetric matrix. Hence we introduce $\Psi \in so(3)$ defined as

$$\Psi = (\Theta - \mathbb{I}_3)(\Theta + \mathbb{I}_3)^{-1}, \quad (2.13)$$

and thus we have shown that almost every element in $SO(3)$ (excluding a small set) can be mapped to $so(3)$.

On the other hand, for every $\tilde{\Psi} \in so(3)$, we can define $\tilde{\Theta} = (\mathbb{I}_3 - \tilde{\Psi})^{-1}(\mathbb{I}_3 + \tilde{\Psi})$ (as the eigen-values of $\tilde{\Psi}$ are pure imaginary). As $\tilde{\Theta}$ doesn't have any eigen value at -1, $\tilde{\Psi}$ can be represented as $\tilde{\Psi} = (\tilde{\Theta} - \mathbb{I}_3)(\tilde{\Theta} + \mathbb{I}_3)^{-1}$. Then for any $x \in \mathbb{R}^3$, we have

$$\langle x, (\tilde{\Theta} - \mathbb{I}_3)(\tilde{\Theta} + \mathbb{I}_3)^{-1}x \rangle = 0. \quad (2.14)$$

Now we introduce $z = (\tilde{\Theta} + \mathbb{I}_3)^{-1}x$, and this allows us to express (2.14) as

$$\begin{aligned} \langle (\tilde{\Theta} + \mathbb{I}_3)z, (\tilde{\Theta} - \mathbb{I}_3)z \rangle &= 0 \quad \text{for any } z \in \mathbb{R}^3, \\ \text{or equivalently, } \tilde{\Theta}^T \tilde{\Theta} &= \mathbb{I}_3. \end{aligned} \quad (2.15)$$

Moreover, we have $\det(\tilde{\Theta}) = 1$. Thus we can conclude that the map

$$\begin{aligned} f &: so(3) \rightarrow SO(3) \\ \Psi &\mapsto (\mathbb{I}_3 - \Psi)^{-1}(\mathbb{I}_3 + \Psi) \end{aligned} \tag{2.16}$$

is injective, but not surjective. A thin set U on $SO(3)$, defined as $U \triangleq \{\Theta \in SO(3) : \det(\Theta + \mathbb{I}_3) = 0\}$, does not have any inverse image under this f . Thus the parametrization of $SO(3)$ using Cayley transform does not suffer from singularity issues. As each element in $so(3)$ can be identified with an element in \mathbb{R}^3 (through the inverse *hat* operator) and vice-versa, we can reformulate an optimization problem over $SO(3) \setminus U$ as an optimization problem over \mathbb{R}^3 .

2.2.3 Customization for Mathematical Programming

As the speed ($\nu(\cdot) = |\dot{\mathbf{r}}(\cdot)|$) is allowed only to be positive, we introduce a new variable $\tilde{\nu}$ defined as $\tilde{\nu}(\cdot) = \ln(\nu(\cdot))$. In other words $\nu(\cdot) = e^{\tilde{\nu}(\cdot)}$. This parametrization along with the usage of Cayley transform and assumption for piecewise constant control inputs allows us to transform the original optimization problem (2.1) into an equivalent optimization problem over \mathbb{R}^l , where l is a large integer of appropriate value. Now we assume uniform sampling of the trajectory and let $[t_{k-1}, t_k]$ be partitioned into M equal sub intervals, i.e. $t_k - t_{k-1} = M\delta$ where δ is defined as $\delta = \frac{t_N - t_0}{MN}$. We define \mathcal{U} to be the sequence of piecewise constant curvatures $\{u_1, u_2, u_3, \dots, u_{NM}\}$. In a similar fashion, \mathcal{V} and Ξ represent the sequences of piecewise controls $\{v_1, v_2, v_3, \dots, v_{NM}\}$ and $\{\tilde{\nu}_1, \tilde{\nu}_2, \tilde{\nu}_3, \dots, \tilde{\nu}_{NM}\}$, respectively. Now, by

letting $\check{R}(0) \in \mathbb{R}^3$ define the initial orientation of the natural Frenet frame through Cayley transform, the equivalent optimization problem can be written as:

$$\begin{aligned} & \min_{\mathcal{U} \in \mathbb{R}^{NM}, \mathcal{V} \in \mathbb{R}^{NM}, \Xi \in \mathbb{R}^{NM}} \left(\sum_{j=0}^N \mathcal{C}_j \right) \quad (2.17) \\ & \check{R}(0) \in \mathbb{R}^3, \mathbf{r}(0) \in \mathbb{R}^3 \end{aligned}$$

subject to the constraints

$$\begin{aligned} R(t_k) &= R(t_{k-1}) \begin{bmatrix} \cos(\delta_k \alpha_k e^{\tilde{v}_k}) & -\frac{u_k}{\alpha_k} \sin(\delta_k \alpha_k e^{\tilde{v}_k}) & -\frac{v_k}{\alpha_k} \sin(\delta_k \alpha_k e^{\tilde{v}_k}) \\ \frac{u_k}{\alpha_k} \sin(\delta_k \alpha_k e^{\tilde{v}_k}) & \frac{(v_k^2 + u_k^2 \cos(\delta_k \alpha_k e^{\tilde{v}_k}))}{\alpha_k^2} & \frac{u_k v_k (\cos(\delta_k \alpha_k e^{\tilde{v}_k}) - 1)}{\alpha_k^2} \\ \frac{v_k}{\alpha_k} \sin(\delta_k \alpha_k e^{\tilde{v}_k}) & \frac{u_k v_k (\cos(\delta_k \alpha_k e^{\tilde{v}_k}) - 1)}{\alpha_k^2} & \frac{(u_k^2 + v_k^2 \cos(\delta_k \alpha_k e^{\tilde{v}_k}))}{\alpha_k^2} \end{bmatrix} \\ \mathbf{r}(t_k) &= \mathbf{r}(t_{k-1}) + \frac{1}{\alpha_k^2} \left[\alpha_k \sin(\delta_k \alpha_k e^{\tilde{v}_k}) \mathbf{x}(t_{k-1}) + u_k (1 - \cos(\delta_k \alpha_k e^{\tilde{v}_k})) \mathbf{y}(t_{k-1}) \right. \\ & \quad \left. + v_k (1 - \cos(\delta_k \alpha_k e^{\tilde{v}_k})) \mathbf{z}(t_{k-1}) \right] \end{aligned}$$

where $\alpha_k = \sqrt{u_k^2 + v_k^2}$. Moreover, the cost associated with each interval is given by

$$\mathcal{C}_j = \|\mathbf{r}(t_j) - r_j\|^2 + \lambda \sum_{i=(j-1)M+1}^{jM} \frac{1}{\delta} \left[(u_i - u_{i-1})^2 + (v_i - v_{i-1})^2 + e^{2\tilde{v}_i} (\tilde{v}_i - \tilde{v}_{i-1})^2 \right]$$

for any $j \in \{1, 2, 3, \dots, N-1, N\}$ and

$$\mathcal{C}_0 = \|\mathbf{r}(t_0) - r_0\|^2.$$

Thus, as a consequence of these numerical adjustments (reformulation over a restricted search space of piecewise constant functions) and various reparametrizations (Cayley transform, exponential function), we end up solving an optimization problem over \mathbb{R}^{3NM+6} .

2.2.4 Multi-stage Approach for Optimization

We adopt a *multi resolution approach* for achieving faster convergence in the optimization problem. In this approach, we first sample the data points, i.e. r_j 's at a coarse resolution and run the optimization routine to yield a better control input. Then we use the midpoint rule to interpolate the optimized control inputs and use this finer set as an initial search point for the next step optimization with higher resolution. Another important fact requires attention while going into a finer data set. If the finer data set consisted of even number of data points, then we have to include an extra data point at the end. The curvature and speed value from the last interval of the coarse data set are extended to handle this situation. This process has been explained pictorially in Fig 3. We keep on repeating this process until all the data points are used.

Once all the data points are taken into consideration we focus our attention to the missing data points¹ within a trajectory, and by introducing some virtual points (as shown in Fig.(4)) we attempt to have uniformly sampled curvature and speed data. These set of virtual points equipartition the whole duration of the

¹As three-dimensional position data, in most of the cases, is obtained via stereo-reconstruction from multiple planar images, occlusion gives rise to missing data points in the raw dataset. Also, *leaving-out-one* strategy for OCV is another source of missing data points.

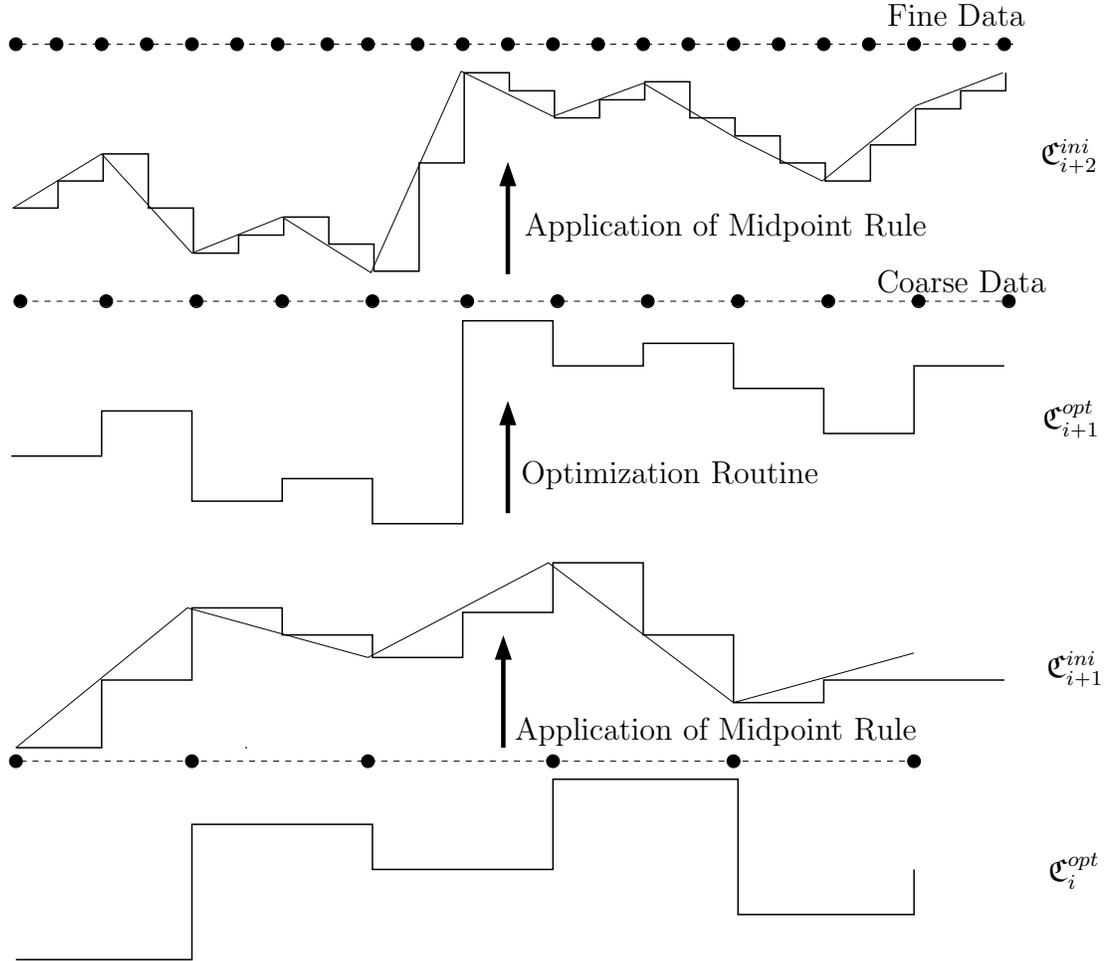


Figure 3: This figure illustrates the multi resolution approach used in solving the optimization problem (2.17). The original data set (with 23 data points) is downsampled twice to yield a coarse data set with 6 data points. Once the control inputs are optimized for the coarse data set, we apply mid-point rule, along with extrapolation, to generate the initial search points for the intermediate resolution data set with 12 data points. Then, after optimization has been carried out for this stage, we apply mid-point rule to obtain the initial search points for the original data set.

trajectory without incurring any extra fitting cost, but we consider the smoothness cost associated with them. As a result the optimization yields better result.

We also proceed beyond the given resolution by dividing the interval between two consecutive data points (real or virtual) into smaller sub-intervals, i.e. our approach is capable of up-sampling. Thus we obtain more finely interpolated values of speed (ν) and curvatures (u and v).

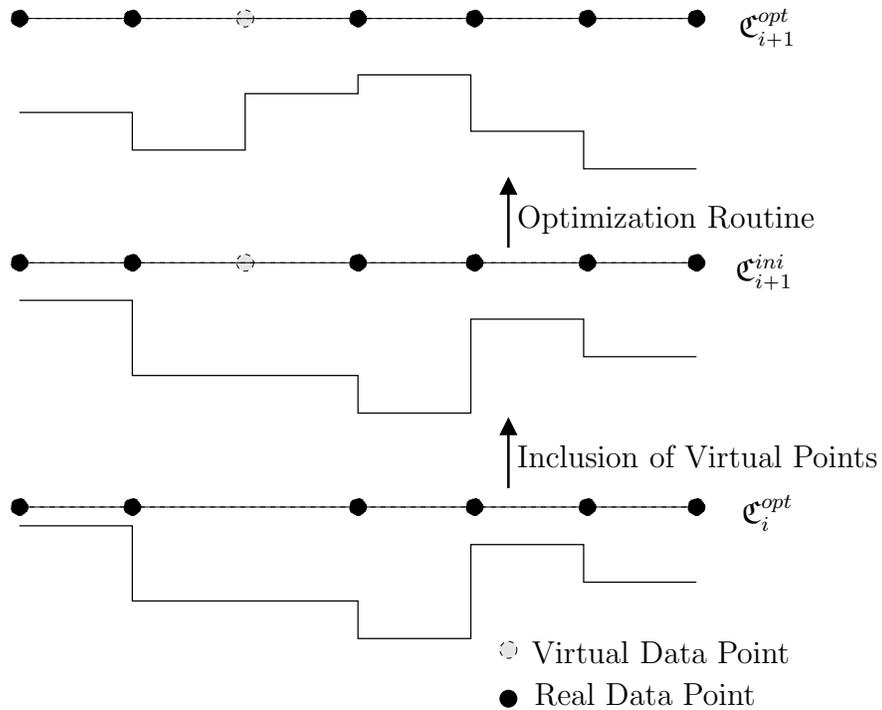


Figure 4: This figure illustrates the inclusion of virtual points to take care of the missing data points.

2.3 Ordinary Cross Validation for the Regularized Inversion Problem

Ordinary cross validation (OCV) was first proposed by Allen [1974] (in the context of regression) and Wahba & Wold [1975] (for smoothing splines). The main idea behind cross validation is to use a subset of the given dataset to obtain a parameter estimate and to use the rest of the data to validate the performance under that estimate. However, cross validation does not use one subset solely for one purpose (estimation or validation); it allows each data point to be used for both purposes. For instance, we can divide the data set into m subsets; compute an estimate from all the subsets but one; and validate the estimate using the left-out subset. Then, we perform the estimation-validation after leaving out a different

subset. This process is repeated multiple times until every possible subset has been considered for validation. In our work we use “leaving-out-one” strategy for perform OCV. Here, an estimate for the trajectory is obtained using all but one data points (by solving an optimization problem), and then the prediction error is computed at the left out data point. Once this process has been repeated for each data point, the prediction errors are summed to yield the ordinary cross validation cost, and an optimal value of the regularization parameter (λ) is chosen in such a way that it minimizes the OCV cost.

In what follows we provide a brief description of the ordinary cross validation procedure for the regularized inversion problem of our interest (2.1). Let $\mathbf{r}_\lambda^k(\cdot)$ be a minimizer of the following optimization problem:

$$\min \left(\sum_{\substack{j=0 \\ j \neq k}}^N \|\mathbf{r}(t_j) - r_j\|^2 + \lambda \int_{t_0}^{t_N} (\dot{u}^2(\sigma) + \dot{v}^2(\sigma) + \dot{w}^2(\sigma)) d\sigma \right) \quad (2.18)$$

subject to the dynamical constraint given by (2.3), where $g(t)$ and $\xi(t)$ are defined in (2.2) and (2.4), respectively. It is worth mentioning here that we solve this optimization problem (2.18) for ordinary cross validation, by transforming it into an equivalent optimization over a high dimensional Euclidean space with discrete constraints (following the guidelines described in Section 2.2). Then the *ordinary cross validation cost* $V_0(\lambda)$ is defined as

$$V_0(\lambda) = \frac{1}{N+1} \sum_{j=0}^N \|\mathbf{r}_\lambda^j(t_j) - r_j\|^2, \quad (2.19)$$

and the corresponding OCV estimate for λ is given by

$$\lambda^* = \underset{\lambda > 0}{\operatorname{argmin}} (V_0(\lambda)). \quad (2.20)$$

2.4 Numerical Results

As a demonstration of concept for our proposed approach, we devise six toy problems related to reconstruction of a circular helix with radius $r = 0.2$ and pitch $2\pi h$ with $h = 0.25$. Hence the helix can be parametrized as $(r \sin(\omega t), r \cos(\omega t), h\omega t)$ with $\omega = \frac{1}{\sqrt{r^2+h^2}}$ being a speed normalizing factor. 46 equi-spaced noisy observations are made along the length of the helix (from $t_0 = 0$ to $t_N = 3.6$ time units). These six toy problems under consideration vary only in terms of observation noise, which is independent and identically distributed zero mean Gaussian in each of these six cases. The variance varies from problem to problem which is equivalent to a changing the Signal-to-Noise ratio (SNR).

As expected, our numerical experiments show that there is a strong relationship between SNR and an OCV estimate for the smoothing parameter (λ^*) (Ta-

Prob. No.	Noise Std. Deviation (σ)	OCV Estimate of λ (λ^*)	SNR
1	0.005	1.50×10^{-6}	40
2	0.010	2.50×10^{-6}	20
3	0.015	9.00×10^{-6}	13.33
4	0.020	7.50×10^{-6}	10
5	0.040	1.50×10^{-5}	5
6	0.050	2.60×10^{-4}	4

Table 2.1: Variation of OCV estimate of λ with different values of signal-to-noise ratio (SNR).

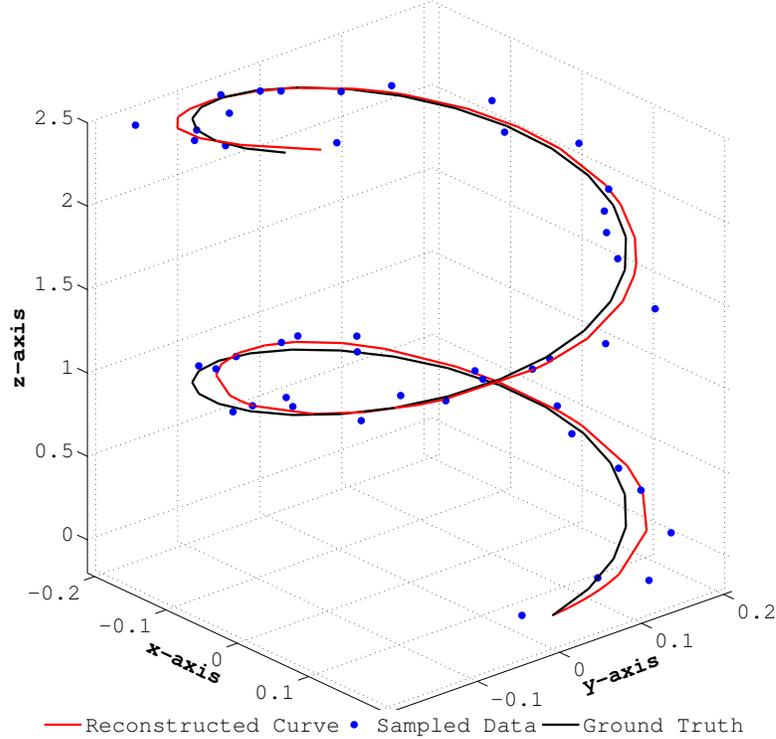


Figure 5: This figure illustrates the reconstruction of a circular helix (standard deviation of the zero-mean observation noise 0.020).

ble 2.1). Moreover, from these observations, we can conclude that λ^* varies almost proportionally with the inverse of SNR for higher values of SNR.

Once we obtain an OCV estimate for the smoothing parameter (λ^*), we reconstruct the trajectory from the observed data points. The reconstructed trajectory for one of the problems is shown in Fig 5.

2.5 Conclusion

In this chapter, we have formalized the inverse problem of trajectory reconstruction. Using a nonlinear generative model, we have defined this data smoothing problem as an optimization problem, and treated it from a mathematical programming perspective. Although lack of integrability refrains us from getting a semi-

analytic solution, we have been able to reformulate the problem over a restricted search space of piecewise constant functions, and then, by exploiting numerical reparametrization (Cayley transform, exponential function), we solved it numerically over a very high dimensional Cartesian space. However, this numerical optimization is non-convex, computationally very demanding, and can at best lead to a local minimum. Finally, we would like to conclude this chapter by mentioning that this algorithm has been applied to reconstruct flight trajectories of bat foraging events (as discussed in Chapter 5).

Chapter 3: Data Smoothing through Linear Quadratic Optimal Control

Earlier we have noticed (in Chapter 2) that lack of an integrability theory prevents us from solving the trajectory reconstruction problem (with natural Frenet frame as the underlying generative model) in an analytic way. Although we have solved it numerically over a restricted search space of piecewise constant functions, this numerical optimization problem is non-convex, computationally very demanding, and can at best lead to a local minimum.

On the other hand, by noting that the steering controls can be expressed in terms of lateral acceleration $\dot{\mathbf{x}}$, we present an alternative linear generative model (in Section 3.1), and exploit the well-developed integrability theory of linear-quadratic optimal control to obtain an analytic solution. Our proposed generative model is fundamentally a triple integrator driven by *jerk*, the derivative of acceleration, as the control input, and we impose regularization by trading total fit-error against high values of the jerk path integral.¹

¹A preliminary version of this work can be found in a previous paper by Dey & Krishnaprasad [2012]. Also, a significant portion of this chapter has been taken verbatim from a pre-print by Dey & Krishnaprasad [2014b].

3.1 Problem Formulation

The primary objective of this work is to reconstruct a trajectory and extract relevant parameters of motion (namely speed, curvatures, *etc.*) from a given time series of observed positions (from motion capture system, GPS data) in a three dimensional space. To ensure smoothness of the reconstructed trajectory we penalize high values of the jerk path integral which is very significant in the context of physiological movement. As described in the literature of locomotion and manipulation (Flash & Hogan [1985]; Todorov & Jordan [1998]), the *2/3-power law*² can be interpreted as a consequence of the minimization of jerk path integral.

Let $\{r_i\}_{i=0}^N$ be the set of observed positions. Then we are interested in finding a trajectory $\mathbf{r} : [t_0, t_N] \rightarrow \mathbb{R}^3$ to minimize the following cost:

$$\sum_{i=0}^N \|\mathbf{r}(t_i) - r_i\|^2 + \lambda \int_{t_0}^{t_N} \|\mathbf{r}^{(3)}(t)\|^2 dt$$

where $(\cdot)^{(k)}$ implies the k -th derivative of a function, if it exists. Similar to the nonlinear optimization problem (2.1) mentioned in Section 2.1, the regularization parameter λ forces a balance between goodness of fit and smoothness of the trajectory. The trajectory dynamics, or in other words the underlying generative model,

²The *power law* says that the speed of an endpoint is inversely proportional to the 1/3-rd power of curvature of the end effector.

is given by

$$\begin{aligned}
 \dot{\mathbf{r}}(t) &= \mathbf{v}(t) \\
 \dot{\mathbf{v}}(t) &= \mathbf{a}(t) \\
 \dot{\mathbf{a}}(t) &= \mathbf{u}(t)
 \end{aligned} \tag{3.1}$$

where $\mathbf{v}(\cdot)$, $\mathbf{a}(\cdot)$ and $\mathbf{u}(\cdot)$ represent the velocity, acceleration and jerk respectively.

Then the cost can be expressed as:

$$\sum_{i=0}^N \|\mathbf{r}(t_i) - r_i\|^2 + \lambda \int_{t_0}^{t_N} \|\mathbf{u}(t)\|^2 dt.$$

Now we define a state-vector $x : [t_0, t_N] \rightarrow \mathbb{R}^9$ as:

$$x \triangleq \begin{bmatrix} \mathbf{r} \\ \mathbf{v} \\ \mathbf{a} \end{bmatrix}. \tag{3.2}$$

Moreover, the input and output of the underlying dynamical system is represented by $u : [t_0, t_N] \rightarrow \mathbb{R}^3$ and $y : [t_0, t_N] \rightarrow \mathbb{R}^3$, respectively. Clearly, $u = \mathbf{u}$ and $y = \mathbf{r}$ for the problem under consideration. Therefore the underlying generative model for a trajectory can be represented in the following compact form,

$$\begin{aligned}
 \dot{x}(t) &= Ax(t) + Bu(t) \\
 y(t) &= Cx(t),
 \end{aligned} \tag{3.3}$$

where

$$A = \begin{bmatrix} 0 & \mathbb{I}_3 & 0 \\ 0 & 0 & \mathbb{I}_3 \\ 0 & 0 & 0 \end{bmatrix}; B = \begin{bmatrix} 0 \\ 0 \\ \mathbb{I}_3 \end{bmatrix}; C = \begin{bmatrix} \mathbb{I}_3 & 0 & 0 \end{bmatrix}. \quad (3.4)$$

It is obvious that the system dynamics (3.3) is both controllable and observable.

Now we can pose our trajectory reconstruction problem as a special case of the following linear-quadratic optimal control problem with intermediate state costs:

$$\begin{aligned} \text{Minimize}_{x(t_0), u} \quad & J(x(t_0), u) = \sum_{i=0}^N \|y(t_i) - r_i\|^2 + \lambda \int_{t_0}^{t_N} u^T(t)u(t)dt \\ \text{subject to} \quad & \text{System dynamics (3.3),} \end{aligned} \quad (3.5)$$

$$x(t_0) \in \mathbb{R}^9, \quad u \in \mathcal{U},$$

where \mathcal{U} is the space of real-valued functions defined on the interval $[t_0, t_N]$.

Now we establish a relationship between the linear (3.1) and non-linear (1.1) generative models for trajectory evolution. First we show that the velocity, acceleration and jerk of a trajectory can be expressed in terms of the parameters of motion obtained from the nonlinear generative model (2.1), namely speed, curvatures and the frame vectors.

$$\left. \begin{aligned} \mathbf{v} &= \nu \mathbf{x} \\ \mathbf{a} &= \dot{\nu} \mathbf{x} + u\nu^2 \mathbf{y} + v\nu^2 \mathbf{z} \\ \mathbf{u} &= (\ddot{\nu} - \nu^3(u^2 + v^2))\mathbf{x} + (3u\nu\dot{\nu} + \dot{u}\nu^2)\mathbf{y} + (3v\nu\dot{\nu} + \dot{v}\nu^2)\mathbf{z}. \end{aligned} \right\} \quad (3.6)$$

From (3.6) it becomes clear that the particular penalty term in (3.5) carries a similar,

although not the same, effect as the penalty term considered in (2.1). Alternatively, speed and curvatures can be expressed in terms of velocity, acceleration and jerk of the trajectory, and it enables us to use the results of (3.5) for curvature based analysis of strategies and steering control laws. This inverse map is given by,

$$\left. \begin{aligned} \nu &= \|\mathbf{v}\| \\ \mathbf{x} &= \frac{\mathbf{v}}{\|\mathbf{v}\|} \\ \dot{\mathbf{x}} &= \frac{1}{\nu} (\mathbf{a} - (\mathbf{a} \cdot \mathbf{x})\mathbf{x}) \\ \kappa &= \frac{\|\dot{\mathbf{x}}\|}{\nu} \\ \tau &= \frac{\mathbf{v} \cdot (\mathbf{a} \times \mathbf{u})}{\|\mathbf{v} \times \mathbf{a}\|^2}, \end{aligned} \right\} \quad (3.7)$$

where κ and τ denote the classical curvature and torsion of the trajectory, respectively. One can view (3.6) and (3.7) as a dictionary between two alternative viewpoints for trajectory generation.

3.2 A Control Theoretic Approach

As λ has a positive value, the constrained optimization in (3.5) can be viewed as a relaxed version of the well-studied fixed endpoint optimal control problem. We begin by applying a standard tool from the theory of least squares, namely the path independence lemma for trajectories of linear systems [Brockett, 1970]. Now onwards, we will not show explicit time dependence for brevity of notation wherever doing so does not create any ambiguity.

In this section we consider a broader class of generative models whose dynamics

is governed by a linear time invariant system (3.3), with $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$ and $C \in \mathbb{R}^{p \times n}$ (i.e. an m -input, p -output system with n -states). We only assume that the pair $[A, B]$ is controllable and the pair $[A, C]$ is observable.

3.2.1 Path Independence Lemma and Its Application to Data Smoothing

Consider the quadratic form $x^T(t)K(t)x(t)$, where $K : [t_0, t_N] \rightarrow \mathbb{R}^{n \times n}$ is a symmetric matrix-valued function. Then, along any trajectory of the underlying dynamical system (3.5), we have

$$\begin{aligned}
& \int_{t_i^+}^{t_{i+1}^-} d(x^T K x) = \int_{t_i^+}^{t_{i+1}^-} \left(x^T K (Ax + Bu) + (Ax + Bu)^T K x + x^T \dot{K} x \right) dt \\
\Rightarrow & \int_{t_i^+}^{t_{i+1}^-} \begin{bmatrix} x \\ u \end{bmatrix}^T \begin{bmatrix} A^T K + KA + \dot{K} & KB \\ B^T K & 0 \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} dt \\
& + x^T(t_i)K(t_i^+)x(t_i) - x^T(t_{i+1})K(t_{i+1}^-)x(t_{i+1}) = 0. \tag{3.8}
\end{aligned}$$

Adding (3.8) over $(t_0^+, t_1^-), \dots, (t_{N-1}^+, t_N^-)$ we obtain

$$\begin{aligned}
& \int_{t_0}^{t_N} \begin{bmatrix} x \\ u \end{bmatrix}^T \begin{bmatrix} A^T K + KA + \dot{K} & KB \\ B^T K & 0 \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} dt + x^T(t_0)K(t_0^-)x(t_0) \\
& + \sum_{i=0}^N x^T(t_i) (K(t_i^+) - K(t_i^-)) x(t_i) - x^T(t_N)K(t_N^+)x(t_N) = 0. \tag{3.9}
\end{aligned}$$

As the quantity given by (3.9) equals to zero for any $u \in \mathcal{U}$ and any K differentiable over $(t_i^+, t_{i+1}^-) \forall i \in \{0, 1, 2, \dots, N-1\}$, a multiple of it can be added to the cost $J(x(t_0), u)$ without incorporating further changes. Hence we have,

$$\begin{aligned}
J(x(t_0), u) &= \lambda x^T(t_0)K(t_0^-)x(t_0) - \lambda x^T(t_N)K(t_N^+)x(t_N) + \sum_{i=0}^N (r_i^T r_i - 2x^T(t_i)C^T r_i) \\
&\quad + \sum_{i=0}^N x^T(t_i) [\lambda (K(t_i^+) - K(t_i^-)) + C^T C] x(t_i) \\
&\quad + \lambda \int_{t_0}^{t_N} \begin{bmatrix} x \\ u \end{bmatrix}^T \begin{bmatrix} A^T K + KA + \dot{K} & KB \\ B^T K & \mathbb{I}_m \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} dt. \tag{3.10}
\end{aligned}$$

As (3.10) holds true for any choice of K , at this point we make the following assumptions on K ,

$$\begin{aligned}
\dot{K}(t) &= -A^T K(t) - K(t)A + K(t)BB^T K(t), \\
K(t_N^+) &= 0, \\
K(t_i^+) - K(t_i^-) &= -\frac{1}{\lambda} C^T C.
\end{aligned} \tag{3.11}$$

With these assumptions (3.11), the cost $J(x(t_0), u)$ can be represented as

$$\begin{aligned}
J(x(t_0), u) &= \lambda x^T(t_0)K(t_0^-)x(t_0) + \sum_{i=0}^N (r_i^T r_i - 2x^T(t_i)C^T r_i) \\
&\quad + \lambda \int_{t_0}^{t_N} \|B^T K(t)x(t) + u(t)\|^2 dt. \tag{3.12}
\end{aligned}$$

Now consider the linear functional $x^T(t)\eta(t)$, where $\eta : [t_0, t_N] \rightarrow \mathbb{R}^n$ is a vector valued function. Then,

$$\begin{aligned} & \int_{t_i^+}^{t_{i+1}^-} d(x^T \eta) = \int_{t_i^+}^{t_{i+1}^-} (x^T \dot{\eta} + (Ax + Bu)^T \eta) dt \\ \Rightarrow & \int_{t_i^+}^{t_{i+1}^-} \begin{bmatrix} x \\ u \end{bmatrix}^T \begin{bmatrix} A^T \eta + \dot{\eta} \\ B^T \eta \end{bmatrix} dt + x^T(t_i) \eta(t_i^+) - x^T(t_{i+1}) \eta(t_{i+1}^-) = 0. \end{aligned} \quad (3.13)$$

Adding (3.13) over $(t_0^+, t_1^-), (t_1^+, t_2^-), \dots, (t_{N-1}^+, t_N^-)$ we obtain

$$\begin{aligned} & \int_{t_0}^{t_N} \begin{bmatrix} x \\ u \end{bmatrix}^T \begin{bmatrix} A^T \eta + \dot{\eta} \\ B^T \eta \end{bmatrix} dt + x^T(t_0) \eta(t_0^-) + \sum_{i=0}^N x^T(t_i) (\eta(t_i^+) - \eta(t_i^-)) \\ & - x^T(t_N) \eta(t_N^+) = 0. \end{aligned} \quad (3.14)$$

As the quantity given by (3.14) equals to zero for any $u \in \mathcal{U}$ and any η differentiable over $(t_i^+, t_{i+1}^-) \forall i \in \{0, 1, 2, \dots, N-1\}$, a multiple of it can be added to the cost $J(x(t_0), u)$ in (3.12) without causing any change. Hence we have,

$$\begin{aligned} J(x(t_0), u) &= \lambda (x^T(t_0) K(t_0^-) x(t_0) + x^T(t_0) \eta(t_0^-)) - \lambda x^T(t_N) \eta(t_N^+) \\ &+ \sum_{i=0}^N x^T(t_i) [\lambda (\eta(t_i^+) - \eta(t_i^-)) - 2C^T r_i] + \sum_{i=0}^N r_i^T r_i \\ &+ \lambda \int_{t_0}^{t_N} \left(\begin{bmatrix} x \\ u \end{bmatrix}^T \begin{bmatrix} A^T \eta + \dot{\eta} \\ B^T \eta \end{bmatrix} + \|B^T Kx + u\|^2 \right) dt. \end{aligned} \quad (3.15)$$

As (3.15) holds true for any choice of η , we make the following assumptions on η ,

$$\begin{aligned}\dot{\eta}(t) &= - (A^T - K(t)BB^T) \eta(t), \\ \eta(t_N^+) &= 0, \\ \eta(t_i^+) - \eta(t_i^-) &= \frac{2}{\lambda} C^T r_i.\end{aligned}\tag{3.16}$$

With these assumptions (3.16), the cost $J(x(t_0), u)$ can be represented as

$$\begin{aligned}J(x(t_0), u) &= \lambda [x^T(t_0)K(t_0^-)x(t_0) + x^T(t_0)\eta(t_0^-)] - \frac{\lambda}{4} \int_{t_0}^{t_N} \|B^T \eta(t)\|^2 dt \\ &\quad + \sum_{i=0}^N r_i^T r_i + \lambda \int_{t_0}^{t_N} \|u(t) + B^T \left(K(t)x(t) + \frac{1}{2}\eta(t) \right)\|^2 dt.\end{aligned}\tag{3.17}$$

From (3.17) it is clear, that by choosing

$$u(t) = u_{opt}(t) \triangleq -B^T \left(K(t)x(t) + \frac{1}{2}\eta(t) \right),\tag{3.18}$$

we have

$$J(x(t_0), u_{opt}) = \lambda (x^T(t_0)K(t_0^-)x(t_0) + x^T(t_0)\eta(t_0^-)) + \sum_{i=0}^N r_i^T r_i - \frac{1}{4}\lambda \int_{t_0}^{t_N} \|B^T \eta(t)\|^2 dt.\tag{3.19}$$

As $\lambda > 0$, it is apparent from (3.19), that the necessary and sufficient condition for the cost to be minimized is,

$u = u_{opt}$ **and** $x^T(t_0)K(t_0^-)x(t_0) + x^T(t_0)\eta(t_0^-)$ be minimized over $x(t_0) \in \mathbb{R}^n$.

Therefore,

$$x_{opt}(t_0) = \arg \min_{x(t_0) \in \mathbb{R}^n} (x^T(t_0)K(t_0^-)x(t_0) + x^T(t_0)\eta(t_0^-)),$$

or in other words, the optimal initial state satisfies the following condition

$$[K(t_0^-)] x_{opt}(t_0) + \frac{1}{2}\eta(t_0^-) = 0. \quad (3.20)$$

Hence, we have

$$J_{min} = \sum_{i=0}^N r_i^T r_i - \lambda \left[x_{opt}^T(t_0)K(t_0^-)x_{opt}(t_0) + \frac{1}{4} \int_{t_0}^{t_N} \|B^T \eta(t)\|^2 dt \right]. \quad (3.21)$$

It is clear from definition (3.5) that the cost is never negative, and hence

$$J_{min} \geq 0$$

or in other words

$$x_{opt}^T(t_0)K(t_0^-)x_{opt}(t_0) + \frac{1}{4} \int_{t_0}^{t_N} \|B^T \eta(t)\|^2 dt \leq \frac{1}{\lambda} \sum_{i=0}^N r_i^T r_i.$$

Proposition 3.1 (Brockett [1970]). *A Riccati equation of the form*

$$\dot{K}(t) = -K(t)A - A^T K(t) + K(t)BB^T K(t) \quad (3.22)$$

$$K(T) = Q,$$

has a symmetric, positive semi-definite solution $K(t)$ for $t \leq T$ whenever the termi-

nal value Q is symmetric, positive semi-definite, and the pair $[A, B]$ is controllable.

Moreover, the solution can be represented as

$$K(t) = e^{-A^T(t-T)} \left(Q - Q \left[(\mathbf{G}(t, T))^{-1} + Q \right]^{-1} Q \right) e^{-A(t-T)} \quad (3.23)$$

where $\mathbf{G}(t, T)$ is a controllability Grammian-like quantity.

Proof for Proposition.3.1. The adjoint system corresponding to the Riccati equation

(3.22) is

$$\frac{d}{dt} \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} = \begin{bmatrix} A & -BB^T \\ 0 & -A^T \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix}$$

and the associated transition matrix can be represented as

$$\begin{bmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{bmatrix} (t, T) = \phi(t, T) = \begin{bmatrix} e^{A(t-T)} & e^{A(t-T)} \mathbf{G}(t, T) \\ 0 & e^{-A^T(t-T)} \end{bmatrix}$$

where,

$$\mathbf{G}(t, T) \triangleq - \int_T^t e^{A(T-\sigma)} BB^T e^{A^T(T-\sigma)} d\sigma$$

is positive definite for any $t < T$ because of controllability of the pair $[A, B]$.

Hence the solution for (3.22) can be represented as,

$$\begin{aligned} K(t) &= \left[\phi_{21}(t, T) + \phi_{22}(t, T)Q \right] \left[\phi_{11}(t, T) + \phi_{12}(t, T)Q \right]^{-1} \\ &= e^{-A^T(t-T)} Q \left[e^{A(t-T)} (\mathbf{I}_n + \mathbf{G}(t, T)Q) \right]^{-1} \\ &= e^{-A^T(t-T)} Q \left(\mathbf{I}_n + \mathbf{G}(t, T)Q \right)^{-1} e^{-A(t-T)}. \end{aligned}$$

Now by applying the matrix inversion lemma,

$$(E - FH^{-1}G)^{-1} = E^{-1} + E^{-1}F(H - GE^{-1}F)^{-1}GE^{-1},$$

and by letting $E = \mathbb{I}_n$, $F = -\mathbb{I}_n$, $G = Q$ and $H = (\mathbb{G}(t, T))^{-1}$ we obtain

$$\left(\mathbb{I}_n + \mathbb{G}(t, T)Q\right)^{-1} = \mathbb{I}_n - \left((\mathbb{G}(t, T))^{-1} + Q\right)^{-1}Q.$$

As $\mathbb{G}(t, T) > 0$ for any $t < T$, its inverse is also positive definite for any $t < T$.

Then positive definiteness of $\left[(\mathbb{G}(t, T))^{-1} + Q\right]^{-1}$ is directly implied from the fact that the terminal condition Q is positive semi-definite.

By defining $\mathbb{M}(t) \triangleq (\mathbb{G}(t, T))^{-1}$, we have

$$K(t) = e^{-A^T(t-T)}Q\left[Q - Q[\mathbb{M}(t) + Q]^{-1}Q\right]Qe^{-A(t-T)}.$$

As $\mathbb{M} > 0$ (implicit dependency on time t is not shown for the sake of clarity) and

$Q = Q^T \geq 0$, there exists a non-singular matrix P such that

$$P^TQP = \Lambda$$

$$P^T\mathbb{M}P = \mathbb{I}_n$$

where Λ is a diagonal matrix with non-negative entries. With the above expressions

from simultaneous diagonalization, we have

$$Q - Q[\mathbf{M} + Q]^{-1}Q = (P^T)^{-1} \left[\Lambda - \Lambda(\mathbb{I}_n + \Lambda)^{-1}\Lambda \right] P^{-1}. \quad (3.24)$$

Now, by assuming $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$, $\lambda_i \geq 0$, we obtain

$$\begin{aligned} (\mathbb{I}_n + \Lambda)^{-1} &= \text{diag}\left(\frac{1}{1 + \lambda_1}, \dots, \frac{1}{1 + \lambda_n}\right) \\ \Rightarrow \Lambda(\mathbb{I}_n + \Lambda)^{-1}\Lambda &= \text{diag}\left(\frac{\lambda_1^2}{1 + \lambda_1}, \dots, \frac{\lambda_n^2}{1 + \lambda_n}\right) \\ \Rightarrow \Lambda - \Lambda(\mathbb{I}_n + \Lambda)^{-1}\Lambda &= \text{diag}\left(\frac{\lambda_1}{1 + \lambda_1}, \dots, \frac{\lambda_n}{1 + \lambda_n}\right). \end{aligned}$$

Therefore $\Lambda - \Lambda(\mathbb{I}_n + \Lambda)^{-1}\Lambda$ is a positive semi-definite diagonal matrix, and hence from (3.24), $Q - Q[\mathbf{M} + Q]^{-1}Q$ is a symmetric positive semi-definite matrix.

Hence, $K(t)$ is symmetric, positive semi-definite for any $t < T$. \square

With assurance on the existence of solution, we make the following claim regarding the form of the Riccati equation solution.

Lemma 3.2. *The solution of the Riccati equation (3.11) assumes the form*

$$K(t_i^-) = \frac{1}{\lambda} \sum_{k=i}^N \Phi_{\Sigma}(t_i, t_k) C^T C \Phi_{\Sigma}^T(t_i, t_k)$$

for any $i \in \{0, 1, \dots, N\}$ where $\Sigma(t) = -(A - \frac{1}{2}BB^T K(t))^T$ and Φ_{Σ} is the transition matrix for Σ .

Proof for Lemma.3.2. We will use mathematical induction to prove the above claim.

From the boundary and jump conditions in (3.11) it is obvious that the claim holds

true for $i = N$.

Now we assume that it holds true for $i = m + 1$, or in other words

$$K(t_{m+1}^-) = \frac{1}{\lambda} \sum_{k=m+1}^N \Phi_{\Sigma}(t_{m+1}, t_k) C^T C \Phi_{\Sigma}^T(t_{m+1}, t_k).$$

Using uniqueness of solution, one can easily verify that

$$K(t) = \Phi_{\Sigma}(t, t_{m+1}) K(t_{m+1}^-) \Phi_{\Sigma}^T(t, t_{m+1})$$

satisfies the Riccati differential equation

$$\dot{K}(t) = -A^T K(t) - K(t) A + K(t) B B^T K(t)$$

for any $t \in (t_m, t_{m+1})$.

Therefore,

$$\begin{aligned} K(t_m^-) &= K(t_m^+) + \frac{1}{\lambda} C^T C \\ &= \Phi_{\Sigma}(t_m, t_{m+1}) K(t_{m+1}^-) \Phi_{\Sigma}^T(t_m, t_{m+1}) + \frac{1}{\lambda} C^T C \\ &= \frac{1}{\lambda} \sum_{k=m+1}^N \Phi_{\Sigma}(t_m, t_k) C^T C \Phi_{\Sigma}^T(t_m, t_k) + \frac{1}{\lambda} \Phi_{\Sigma}(t_m, t_m) C^T C \Phi_{\Sigma}^T(t_m, t_m) \\ &= \frac{1}{\lambda} \sum_{k=m}^N \Phi_{\Sigma}(t_m, t_k) C^T C \Phi_{\Sigma}^T(t_m, t_k). \end{aligned}$$

Hence the claim is proved, as it holds true for $i = m$. □

Now we concentrate on the dynamics of η given by (3.16) and introduce a new

time-varying matrix

$$\tilde{\Sigma}(t) = -(A - BB^TK(t))^T.$$

Then the dynamics of η can be represented as

$$\dot{\eta}(t) = \tilde{\Sigma}(t)\eta(t) \tag{3.25}$$

for any $t \in (t_i, t_{i+1}), i \in \{0, 1, \dots, N-1\}$. By letting $\Phi_{\tilde{\Sigma}}$ denote the transition matrix for (3.25), we can make the following claim regarding the solution for η variables.

Lemma 3.3.

$$\begin{aligned} \eta(t_i^+) &= -\frac{2}{\lambda} \sum_{k=i+1}^N \Phi_{\tilde{\Sigma}}(t_i, t_k) C^T r_k \\ \eta(t_i^-) &= -\frac{2}{\lambda} \sum_{k=i}^N \Phi_{\tilde{\Sigma}}(t_i, t_k) C^T r_k \end{aligned}$$

Proof for Lemma.3.3. We will use mathematical induction to prove the above claim.

From the boundary and jump conditions in (3.16) it is obvious that the claim holds true for $i = N$ as,

$$\begin{aligned} \eta(t_N^+) &= 0 \\ \eta(t_N^-) &= -\frac{2}{\lambda} C^T r_N. \end{aligned}$$

Now we assume that it holds true for $i = m + 1$, or in other words

$$\begin{aligned}\eta(t_{m+1}^+) &= -\frac{2}{\lambda} \sum_{k=m+2}^N \Phi_{\tilde{\Sigma}}(t_{m+1}, t_k) C^T r_k \\ \eta(t_{m+1}^-) &= -\frac{2}{\lambda} \sum_{k=m+1}^N \Phi_{\tilde{\Sigma}}(t_{m+1}, t_k) C^T r_k.\end{aligned}$$

Using the dynamics of η , given by (3.25), we have the following relationship

$$\begin{aligned}\eta(t_m^+) &= \Phi_{\tilde{\Sigma}}(t_m, t_{m+1}) \eta(t_{m+1}^-) = -\frac{2}{\lambda} \Phi_{\tilde{\Sigma}}(t_m, t_{m+1}) \sum_{k=m+1}^N \Phi_{\tilde{\Sigma}}(t_{m+1}, t_k) C^T r_k \\ &= -\frac{2}{\lambda} \sum_{k=m+1}^N \Phi_{\tilde{\Sigma}}(t_m, t_k) C^T r_k.\end{aligned}\tag{3.26}$$

Using the jump condition at t_m , we obtain

$$\begin{aligned}\eta(t_m^-) &= \eta(t_m^+) - \frac{2}{\lambda} C^T r_m \\ &= -\frac{2}{\lambda} \sum_{k=m+1}^N \Phi_{\tilde{\Sigma}}(t_m, t_k) C^T r_k - \frac{2}{\lambda} \Phi_{\tilde{\Sigma}}(t_m, t_m) C^T r_m \\ &= -\frac{2}{\lambda} \sum_{k=m}^N \Phi_{\tilde{\Sigma}}(t_m, t_k) C^T r_k.\end{aligned}\tag{3.27}$$

From (3.26) and (3.27) it is clear that the claim holds true for $i = m$.

Hence the claim is proved. \square

Now we focus into the problem of our interest, i.e. the problem of trajectory reconstruction through minimization of the jerk path integral. By exploiting the particular structure of A , B and C (given by (3.4)), namely the triple-integrator property, we claim observability for the $(-\Sigma^T, C)$ pair.

Proposition 3.4. $(-\Sigma^T, C)$ forms an observable pair for the trajectory reconstruction problem (3.3,3.4).

Proof for Proposition.3.4. K is a symmetric matrix by definition, and hence one can assume the following block structure for K ,

$$K(t) = \begin{bmatrix} K_{11}(t) & K_{12}(t) & K_{13}(t) \\ K_{12}^T(t) & K_{22}(t) & K_{23}(t) \\ K_{13}^T(t) & K_{23}^T(t) & K_{33}(t) \end{bmatrix}.$$

With this particular structure for K , we have the following expression of $\Sigma^T(t)$ for the jerk path integral minimization problem,

$$\Sigma^T(t) = \begin{bmatrix} 0 & -\mathbb{I}_3 & 0 \\ 0 & 0 & -\mathbb{I}_3 \\ \frac{1}{2}K_{13}^T(t) & \frac{1}{2}K_{23}^T(t) & \frac{1}{2}K_{33}(t) \end{bmatrix}. \quad (3.28)$$

Now, for the sake of convenience, we use Silverman-Meadows rank condition [Silverman & Meadows, 1967] to prove our claim. To do so, we define the matrix Q_{obv} as

$$Q_{obv}(t) = [S_0(t) \ S_1(t) \ \cdots \ S_{n-1}(t)]$$

where $S_i(t)$'s are computed recursively using

$$S_{k+1}(t) = -\Sigma(t)S_k(t) + \dot{S}_k(t), \quad S_0(t) = C^T. \quad (3.29)$$

The $S_i(t)$'s will assume the following form,

$$S_0(t) = \begin{bmatrix} \mathbb{I}_3 \\ 0 \\ 0 \end{bmatrix}, \quad S_1(t) = \begin{bmatrix} 0 \\ \mathbb{I}_3 \\ 0 \end{bmatrix}, \quad S_2(t) = \begin{bmatrix} 0 \\ 0 \\ \mathbb{I}_3 \end{bmatrix}, \quad S_3(t) = -\frac{1}{2} \begin{bmatrix} K_{13}(t) \\ K_{23}(t) \\ K_{33}(t) \end{bmatrix},$$

and so on. Hence it can be immediately concluded that the pair $(-\Sigma^T, C)$ is observable as the rank of $Q_{obv}(t)$ is 9 for any $t \in \mathbb{R}^+ \cup \{0\}$. \square

Theorem 3.5. *For the trajectory reconstruction problem (3.3,3.4), the optimal initial condition (given by (3.20)) is uniquely solvable for almost any time index set $\{t_i\}_{i=0}^N$.*

Proof for Theorem.3.5. From proposition 1 we have,

$$\begin{aligned} K(t_0^-) &= \frac{1}{\lambda} \sum_{k=0}^N \Phi_{\Sigma}(t_0, t_k) C^T C \Phi_{\Sigma}^T(t_0, t_k) \\ &= \frac{1}{\lambda} \sum_{k=0}^N \Phi_{-\Sigma^T}^T(t_k, t_0) C^T C \Phi_{-\Sigma^T}(t_k, t_0) \\ &= \frac{1}{\lambda} \begin{bmatrix} C \\ C\Phi_{-\Sigma^T}(t_1, t_0) \\ \vdots \\ C\Phi_{-\Sigma^T}(t_N, t_0) \end{bmatrix}^T \begin{bmatrix} C \\ C\Phi_{-\Sigma^T}(t_1, t_0) \\ \vdots \\ C\Phi_{-\Sigma^T}(t_N, t_0) \end{bmatrix} \\ &= \frac{1}{\lambda} \mathfrak{C}^T \mathfrak{C}. \end{aligned}$$

Now we investigate the rank of \mathfrak{C} because the solvability of (3.20) is equivalent to

the fact of \mathfrak{C} having full rank. To do so we consider the following system

$$\begin{aligned}\dot{\xi}(t) &= -\Sigma^T(t)\xi(t) \\ \gamma(t) &= C\xi(t),\end{aligned}\tag{3.30}$$

which is observable (*proposition 3.3*). We can easily show that the j -th derivative of its output can be represented as

$$\gamma^{(j)}(t) = S_j^T(t)\Phi_{-\Sigma^T}(t, t_{ini})\xi(t_{ini})$$

where $S_j(t)$'s are defined in (3.29).

Let $\xi_1 \neq \xi_2$ be two different choice of initial state $\xi(t_0)$ for the system (3.30) and $\gamma_i(t)$ be its output corresponding to the initial condition $\xi(t_0) = \xi_i$. Now we define,

$$\mathcal{Y}_i \triangleq \begin{bmatrix} \gamma_i(t_0) \\ \gamma_i(t_1) \\ \vdots \\ \gamma_i(t_N) \end{bmatrix} = \mathfrak{C}\xi_i.$$

Now we claim that the outputs of (3.30), corresponding to two different initial conditions $\xi_1 \neq \xi_2$, do not match identically over any interval $\mathbb{T} \subset \mathbb{R}^+ \cup \{0\}$, or in other words, there is no such interval $\mathbb{T} \subset \mathbb{R}^+ \cup \{0\}$ such that $\gamma_1(t) = \gamma_2(t)$ for any $t \in \mathbb{T}$.

We can prove our claim by contradiction. Let

$$C\Phi_{-\Sigma^T}(t, t_0)\xi_1 = C\Phi_{-\Sigma^T}(t, t_0)\xi_2$$

for all t belonging to some interval \mathbb{T} . Then the derivatives, when they exist, should match for any t^* in the interior of \mathbb{T} , i.e.

$$\begin{aligned} & \left. \frac{d^j}{dt^j} \left(C\Phi_{-\Sigma^T}(t, t_0)\xi_1 \right) \right|_{t^*} = \left. \frac{d^j}{dt^j} \left(C\Phi_{-\Sigma^T}(t, t_0)\xi_2 \right) \right|_{t^*} \\ \Rightarrow & \begin{bmatrix} S_0^T(t) \\ S_1^T(t) \\ \vdots \\ S_{n-1}^T(t) \end{bmatrix} \Phi_{-\Sigma^T}(t^*, t_0)\xi_1 = \begin{bmatrix} S_0^T(t) \\ S_1^T(t) \\ \vdots \\ S_{n-1}^T(t) \end{bmatrix} \Phi_{-\Sigma^T}(t^*, t_0)\xi_2 \\ \Rightarrow & Q_{obv}^T(t)\Phi_{-\Sigma^T}(t^*, t_0) \left(\xi_1 - \xi_2 \right) = 0 \\ \Rightarrow & \xi_1 = \xi_2. \end{aligned}$$

But it contradicts our initial assumption about inequality of ξ_1 and ξ_2 , thereby proving the claim. Hence $\mathfrak{C}\xi_1 \neq \mathfrak{C}\xi_2$ for almost any time index set $\{t_i\}_{i=0}^N$. Therefore $K(t_0^-)$ is positive definite almost surely because \mathfrak{C} has full rank almost surely.

When the rank condition fails, i.e. $\mathfrak{C}\xi_1 = \mathfrak{C}\xi_2$, we can consider an arbitrary close perturbation of the original time index. For any given $\epsilon > 0$ we can choose a

perturbed time index set $\{\tilde{t}_i\}_{i=0}^N$, such that the following conditions holds true,

$$\begin{aligned}
& t_0 = \tilde{t}_0, \\
& \sum_{i=1}^N |t_i - \tilde{t}_i| < \epsilon, \\
& \text{and, } \begin{bmatrix} C \\ C\Phi_{-\Sigma^T}(\tilde{t}_1, \tilde{t}_0) \\ \vdots \\ C\Phi_{-\Sigma^T}(\tilde{t}_N, \tilde{t}_0) \end{bmatrix} \text{ has full rank.}
\end{aligned}$$

Therefore (3.20) can be uniquely solved, for almost any time index set $\{t_i\}_{i=0}^N$. \square

As $K(t_0^-)$ is shown to be a symmetric, invertible and positive definite matrix, for almost any time index set $\{t_i\}_{i=0}^N$, the optimal initial condition can be represented as

$$x_{opt}(t_0) = -\frac{1}{2} [K(t_0^-)]^{-1} \eta(t_0^-) = \frac{1}{\lambda} [K(t_0^-)]^{-1} \sum_{k=0}^N \Phi_{\Sigma}(t_0, t_k) C^T r_k. \quad (3.31)$$

Remark 3.1. *The work by Magnus Egerstedt, Clyde Martin and their collaborators (Egerstedt & Martin [2010]; Kano et al. [2008]; Shan et al. [2000]; Zhou et al. [2005, 2006]) provides an alternative view for exploiting linear optimal control for smooth interpolation. Their work provides a framework to recover a scalar input from sampled observations of scalar output data by solving a regularized optimal control problem, similar to the one given in (3.5). However, this work uses a variational approach and makes some extra assumption to ensure smoothness. This variational approach can also be viewed from a learning theoretic perspective. On the other hand,*

our approach exploits the integrability of linear-quadratic optimal control problems, and the observations introduce jumps in the optimal control input. Moreover, our results can be used to construct control theoretic splines for multi-input multi-output systems if the optimal initial condition is uniquely solvable from (3.20).

Remark 3.2. *Moreover, our results can be generalized to reconstruct any trajectory whose evolution is governed by a linear time invariant generative model. Given the pair $[A, B]$ ($[A, C]$) is controllable (observable) and the optimal initial condition (x_{opt}) is uniquely solvable from 3.20, one can use our approach for data smoothing. In particular, trajectory reconstruction by penalizing high values of the snap, crackle or pop (4^{th} , 5^{th} or 6^{th} derivative of position, respectively) path integrals, will not affect the structure for higher order integrators, and hence it can be easily shown that Prop 3.4 holds true for those cases. Therefore our approach has a natural extension to tackle trajectory reconstruction through penalizing higher order derivatives of motion.*

3.2.2 Linearity of Reconstruction

Under the action of an optimal control input u_{opt} the system dynamics can be represented as

$$\dot{x}(t) = [A - BB^T K(t)] x(t) - \frac{1}{2} BB^T \eta(t) = -\tilde{\Sigma}^T(t)x(t) - \frac{1}{2} BB^T \eta(t), \quad (3.32)$$

or in other words it can be viewed as a time-varying linear system with η being the input, and the state $x(t)$ can be expressed as

$$\begin{aligned} x(t) &= \Phi_{-\bar{\Sigma}^T}(t, t_0)x_{opt}(t_0) - \frac{1}{2} \int_{t_0}^t \Phi_{-\bar{\Sigma}^T}(t, \sigma)BB^T\eta(\sigma)d\sigma \\ &= \Phi_{\bar{\Sigma}}^T(t_0, t)x_{opt}(t_0) - \frac{1}{2} \int_{t_0}^t \Phi_{\bar{\Sigma}}^T(\sigma, t)BB^T\eta(\sigma)d\sigma. \end{aligned} \quad (3.33)$$

Therefore the reconstructed states can be represented (at sampling time instances $\{t_i\}_{i=0}^N$) as,

$$\begin{aligned} x(t_k) &= \Phi_{\bar{\Sigma}}^T(t_0, t_k)x_{opt}(t_0) - \frac{1}{2} \int_{t_0}^{t_k} \Phi_{\bar{\Sigma}}^T(\sigma, t_k)BB^T\eta(\sigma)d\sigma \\ &= \Phi_{\bar{\Sigma}}^T(t_0, t_k)x_{opt}(t_0) - \frac{1}{2} \sum_{i=1}^k \left[\int_{t_{i-1}}^{t_i} \Phi_{\bar{\Sigma}}^T(\sigma, t_k)BB^T\eta(\sigma)d\sigma \right] \\ &= \Phi_{\bar{\Sigma}}^T(t_0, t_k)x_{opt}(t_0) - \frac{1}{2} \sum_{i=1}^k \left[\int_{t_{i-1}}^{t_i} \Phi_{\bar{\Sigma}}^T(\sigma, t_k)BB^T\Phi_{\bar{\Sigma}}(\sigma, t_i)\eta(t_i^-)d\sigma \right] \\ &= \Phi_{\bar{\Sigma}}^T(t_0, t_k)x_{opt}(t_0) - \frac{1}{2} \sum_{i=1}^k \left[\int_{t_{i-1}}^{t_i} \Phi_{\bar{\Sigma}}^T(\sigma, t_k)BB^T\Phi_{\bar{\Sigma}}(\sigma, t_i)d\sigma \right] \eta(t_i^-) \\ &= \Phi_{\bar{\Sigma}}^T(t_0, t_k)x_{opt}(t_0) \\ &\quad + \frac{1}{\lambda} \sum_{i=1}^k \left[\int_{t_{i-1}}^{t_i} \Phi_{\bar{\Sigma}}^T(\sigma, t_k)BB^T\Phi_{\bar{\Sigma}}(\sigma, t_i)d\sigma \times \left(\sum_{j=i}^N \Phi_{\bar{\Sigma}}(t_i, t_j)C^T r_j \right) \right] \\ &= \Phi_{\bar{\Sigma}}^T(t_0, t_k)x_{opt}(t_0) \\ &\quad + \frac{1}{\lambda} \sum_{i=1}^N \left[\sum_{j=1}^{\min\{i, k\}} \left(\int_{t_{j-1}}^{t_j} \Phi_{\bar{\Sigma}}^T(\sigma, t_k)BB^T\Phi_{\bar{\Sigma}}(\sigma, t_j)d\sigma \right) \times \Phi_{\bar{\Sigma}}(t_j, t_i) \right] C^T r_i. \end{aligned}$$

As the optimal initial condition is linear in observed data points, the smoothed position at time t_k can also be expressed as a linear combination of observed positions.

$$r(t_k) = \frac{1}{\lambda} \sum_{i=0}^N [C\mathcal{F}_\lambda(k, i)C^T] r_i \quad (3.34)$$

where

$$\mathcal{F}_\lambda(k, i) = \Phi_{\hat{\Sigma}}^T(t_0, t_k) [K(t_0^-)]^{-1} \Phi_{\hat{\Sigma}}(t_0, t_i) + \sum_{j=1}^{\min\{i, k\}} \left(\int_{t_{j-1}}^{t_j} \Phi_{\hat{\Sigma}}^T(\sigma, t_k) B B^T \Phi_{\hat{\Sigma}}(\sigma, t_i) d\sigma \right) \quad (3.35)$$

Remark 3.3. *As the coefficients $\mathcal{F}(k, i)$'s depend only on the sampling time instances, namely t_0, \dots, t_N , and the underlying system dynamics, these coefficients can be pre-computed.*

Remark 3.4. *This approach can be perceived as a global alternative to Savitzky-Golay smoothing filters (Savitzky & Golay [1964]; Schafer [2011]), wherein the filtered outputs are obtained by fitting a least square polynomial (locally) through the observed data points. In our approach the local nature is absent, instead each of the filtered outputs depends on the complete data set. But because of this global nature our approach has its own drawback. This method, in its true form, cannot be used in real-time as it requires all the observations together.*

Remark 3.5. *The significance of the word “smoothing” is twofold in this context. Firstly this approach penalizes high values of jerk path integral and thereby yields a smoothed trajectory. Moreover, it uses data from both past and future to estimate the present position and thus justifies the usage of “smoothing” in estimation context.*

Remark 3.6. *The formulation of the problem is an example of fixed interval smoothing. One can use this as a building block and proceed to obtain a fixed lag smoothing algorithm. The path is quite intuitive.*

3.3 An Alternative Co-state Based Approach

Although we have developed an analytic method to solve data smoothing, the procedure is computationally demanding because it involves solving a differential Riccati equation. Now, to make computations more tractable, we represent the solution in terms of co-state variables, defined as,

$$p(t) \triangleq K(t)x(t) + \frac{1}{2}\eta(t). \quad (3.36)$$

Then the optimal control input (3.18) and system dynamics (3.3) will have the form

$$\begin{aligned} u_{opt}(t) &= -B^T p(t) \\ \dot{x}(t) &= Ax(t) - BB^T p(t), \end{aligned}$$

and the dynamics of the co-states is given by

$$\dot{p}(t) = \dot{K}(t)x(t) + K(t)\dot{x}(t) + \frac{1}{2}\dot{\eta}(t) = -A^T p(t). \quad (3.37)$$

Therefore the optimal trajectory between two observation times can be viewed as the base integral curve of the following system

$$\frac{d}{dt} \begin{bmatrix} x(t) \\ p(t) \end{bmatrix} = \begin{bmatrix} A & -BB^T \\ 0 & -A^T \end{bmatrix} \begin{bmatrix} x(t) \\ p(t) \end{bmatrix}. \quad (3.38)$$

From (3.38) it is apparent that the dynamics of p is decoupled from that of x .

Now we'll focus on the jump conditions for the co-states

$$p(t_i^+) - p(t_i^-) = [K(t_i^+) - K(t_i^-)] x(t_i) + \frac{1}{2} [\eta(t_i^+) - \eta(t_i^-)] = \frac{1}{\lambda} C^T (r_i - r(t_i)). \quad (3.39)$$

We also have the following terminal condition

$$p(t_N^+) = K(t_N^+)x(t_N) + \frac{1}{2}\eta(t_N^+) = 0$$

as both $K(t_N^+)$ and $\eta(t_N^+)$ are equal to zero, and by letting $x(t_0) = x_{opt}(t_0)$, (3.20) yields,

$$p(t_0^-) = K(t_0^-)x(t_0) + \frac{1}{2}\eta(t_0^-) = 0.$$

Now we introduce a new variable, namely incremental time, defined as $\Delta_i \triangleq$

$t_{i+1} - t_i$ $i \in \{0, 1, \dots, N-1\}$. From (3.38) we have

$$p(t) = e^{-A^T(t-t_i)} p(t_i^+) \quad t \in (t_i, t_{i+1}) \quad (3.40)$$

$$\begin{aligned} p(t_{i+1}^+) &= p(t_{i+1}^-) + \frac{1}{\lambda} C^T (r_{i+1} - Cx(t_{i+1})) \\ &= e^{-A^T \Delta_i} p(t_i^+) - \frac{1}{\lambda} C^T Cx(t_{i+1}) + \frac{1}{\lambda} C^T r_{i+1} \end{aligned} \quad (3.41)$$

for $i \in \{0, 1, \dots, N-1\}$. From the dynamics of x in (3.38), we have

$$\begin{aligned} x(t_{i+1}) &= e^{A(t_{i+1}-t_i)} x(t_i) - \int_{t_i}^{t_{i+1}} e^{A(t_{i+1}-\sigma)} BB^T p(\sigma) d\sigma \\ &= e^{A \Delta_i} x(t_i) - \int_{t_i}^{t_{i+1}} e^{A(t_{i+1}-\sigma)} BB^T e^{-A^T(\sigma-t_i)} p(t_i^+) d\sigma \\ &= e^{A \Delta_i} x(t_i) - e^{A \Delta_i} \left[\int_{t_i}^{t_{i+1}} e^{A(t_i-\sigma)} BB^T e^{-A^T(\sigma-t_i)} d\sigma \right] p(t_i^+). \end{aligned} \quad (3.42)$$

From (3.41) and (3.42) we obtain the following matrix representation for forward-propagation of $x(t_i)$ and $p(t_i^+)$

$$\begin{aligned} \begin{bmatrix} x(t_{i+1}) \\ p(t_{i+1}^+) \end{bmatrix} &= \begin{bmatrix} e^{A \Delta_i} & -e^{A \Delta_i} W_i \\ -\frac{1}{\lambda} C^T C e^{A \Delta_i} & \left[e^{-A^T \Delta_i} + \frac{1}{\lambda} C^T C e^{A \Delta_i} W_i \right] \end{bmatrix} \begin{bmatrix} x(t_i) \\ p(t_i^+) \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 \\ \frac{1}{\lambda} C^T \end{bmatrix} r_{i+1} \end{aligned} \quad (3.43)$$

where W_i is defined as

$$W_i = \int_{t_i}^{t_{i+1}} e^{A(t_i-\sigma)} B B^T e^{-A^T(\sigma-t_i)} d\sigma = \int_0^{\Delta_i} e^{-A\tau} B B^T e^{-A^T\tau} d\tau \quad (\tau = \sigma - t_i). \quad (3.44)$$

From (3.44) it is apparent that the controllability Gramian W_i depends only on the inter-sample intervals, not explicitly on the sampling instances. Moreover, the Gramian is invertible as the underlying system (3.3) is controllable.

By defining a discrete time state vector as $z_i = [x^T(t_i) p^T(t_i^+)]^T$, (3.43) can be represented as the following discrete time system

$$z_{i+1} = \Lambda_i z_i + \Gamma r_{i+1} \quad (3.45)$$

where Λ_i and Γ are defined as

$$\Lambda_i = \begin{bmatrix} e^{A\Delta_i} & -e^{A\Delta_i} W_i \\ -\frac{1}{\lambda} C^T C e^{A\Delta_i} & \left[e^{-A^T \Delta_i} + \frac{1}{\lambda} C^T C e^{A\Delta_i} W_i \right] \end{bmatrix}$$

$$\Gamma = \begin{bmatrix} 0 \\ \frac{1}{\lambda} C^T \end{bmatrix}.$$

Lemma 3.6. Λ_i is invertible for any $i \in \{0, 1, \dots, N-1\}$.

Proof for Lemma.3.6.

$$\begin{aligned}
\Lambda_i &= \begin{bmatrix} e^{A\Delta_i} & -e^{A\Delta_i}W_i \\ -\frac{1}{\lambda}C^T C e^{A\Delta_i} & \left[e^{-A^T\Delta_i} + \frac{1}{\lambda}C^T C e^{A\Delta_i}W_i \right] \end{bmatrix} \\
&= \begin{bmatrix} \mathbb{I}_n & 0 \\ -\frac{1}{\lambda}C^T C & \mathbb{I}_n \end{bmatrix} \begin{bmatrix} e^{A\Delta_i} & -e^{A\Delta_i}W_i \\ 0 & e^{-A^T\Delta_i} \end{bmatrix} \\
&= M\Upsilon_i
\end{aligned} \tag{3.46}$$

(3.46) gives a block LU-factorization for Λ_i and both M and Υ_i are invertible for any i .

Hence, Λ_i is invertible for any i . □

From (3.45) we obtain

$$\begin{aligned}
z_k &= \left(\prod_{i=0}^{k-1} \Lambda_i \right) \begin{bmatrix} x(t_0) \\ \frac{1}{\lambda}C^T (r_0 - Cx(t_0)) \end{bmatrix} + \sum_{i=1}^k \left(\prod_{j=i}^{k-1} \Lambda_j \right) \Gamma r_i \\
&= \left(\prod_{i=0}^{k-1} \Lambda_i \right) \left(\begin{bmatrix} \mathbb{I}_n \\ -\frac{1}{\lambda}C^T C \end{bmatrix} x(t_0) + \Gamma r_0 \right) + \sum_{i=1}^k \left(\prod_{j=i}^{k-1} \Lambda_j \right) \Gamma r_i \\
&= \left(\prod_{i=0}^{k-1} \Lambda_i \right) \begin{bmatrix} \mathbb{I}_n \\ -\frac{1}{\lambda}C^T C \end{bmatrix} x(t_0) + \sum_{i=0}^k \left(\prod_{j=i}^{k-1} \Lambda_j \right) \Gamma r_i
\end{aligned} \tag{3.47}$$

where \prod represents left multiplication. As $p(t_N^+) = 0$, $x(t_0)$ can be obtained by

solving the following equation

$$[0 \quad \mathbb{I}_n] \left(\prod_{i=0}^{N-1} \Lambda_i \right) \begin{bmatrix} \mathbb{I}_n \\ -\frac{1}{\lambda} C^T C \end{bmatrix} x(t_0) = - [0 \quad \mathbb{I}_n] \sum_{i=0}^N \left(\prod_{j=i}^{N-1} \Lambda_j \right) \Gamma r_i. \quad (3.48)$$

From the way (3.48) has been obtained, it can be inferred that (3.48) is an alternative form of (3.20). Hence, it can be concluded from *Theorem 3.4* that for the trajectory reconstruction problem (3.3,3.4) of our interest, (3.48) yields a unique solution for the optimal initial condition for almost any time index set $\{t_i\}_{i=0}^N$. Once $x(t_0)$ is obtained by solving (3.48), the trajectory can be reconstructed using (3.38) and the jump conditions given by (3.39).

Remark 3.7. *The limiting case of $\lambda = 0$ signifies the exact fitting problem, and hence can be represented as an optimal control problem with both initial and final points lying on an affine space. Although this problem can be solved by applying suitable transversality conditions, it will result in non-unique state trajectories.*

Remark 3.8. *The proposed algorithm for data smoothing is fast, with complexity of the order of sample size ($\mathcal{O}(N)$).*

3.4 Ordinary Cross Validation for Optimal λ Selection

As discussed earlier (in Section 2.3), ordinary cross validation (OCV) performs reconstruction by considering a subset of the whole data set, and then computes the fit-error at left-out data points. After this step has been repeated for all possible subsets, the fit errors are summed up. This sum of errors can be perceived as a

Algorithm 1 Algorithm for trajectory smoothing

Data: Time index - $\{t_i\}_{i=0}^N$; Data - $\{r_i\}_{i=0}^N$; Smoothing Parameter - $\lambda > 0$

Define: A , B and C

for $i = 0$ *to* $N - 1$ **do**

$\Delta t \leftarrow t_{i+1} - t_i$

 Compute the Gramian $W = \int_0^{\Delta t} e^{-A\sigma} B B^T e^{-A^T \sigma} d\sigma$

 — *Due to special structures in A and B , W have a closed form solution (involving polynomial functions of Δt).*

$\Upsilon_i \leftarrow \begin{bmatrix} e^{A\Delta t} & -e^{A\Delta t} W \\ 0 & e^{-A^T \Delta t} \end{bmatrix}$

end

$M \leftarrow \begin{bmatrix} I & 0 \\ -\frac{1}{\lambda} C^T C & I \end{bmatrix}$

$\Gamma \leftarrow \begin{bmatrix} 0 \\ \frac{1}{\lambda} C^T \end{bmatrix}$

Initialize: $P_0 \leftarrow I$

Initialize: $S_0 \leftarrow \Gamma r_0$

for $i = 1$ *to* N **do**

$P_i \leftarrow M * \Upsilon_{i-1} * P_{i-1}$

$S_i \leftarrow M * \Upsilon_{i-1} * S_{i-1} + \Gamma r_i$

end

Define: $\mathbb{A} \leftarrow \begin{bmatrix} 0 & I \end{bmatrix} * P_N * M(:, 1)$

Define: $\mathbb{B} \leftarrow \begin{bmatrix} 0 & -I \end{bmatrix} * S_N$

Solve optimal initial condition: $x_{opt}(t_0) = \mathbb{A}^{-1} \mathbb{B}$.

for $i = 0$ *to* $N - 1$ **do**

$z_i \leftarrow P_i * M(:, 1) * x_{opt}(t_0) + S_i$

$x(t_i) \leftarrow \begin{bmatrix} I & 0 \end{bmatrix} * z_i$

end

Result: Compute reconstructed position - $r(t_i) = CX(t_i)$, $i \in \{0, 1, \dots, N\}$.

sampled variance of the estimator for that particular amount of regularization. Our objective is to pick the amount of regularization (λ -parameter) which minimizes this sample variance. In our case, we have adopted the leaving-out-one strategy for OCV, wherein all-but-one data point is used for reconstruction.

Now we'll briefly discuss the ordinary cross validation procedure for the trajectory smoothing problem. Let $(x_\lambda^k(t_0), u_\lambda^k(\cdot))$ be a minimizer of the following op-

timization problem:

$$\text{Minimize}_{x(t_0), u} \left(\sum_{\substack{j=0 \\ j \neq k}}^N \|y(t_j) - r_j\|^2 + \lambda \int_{t_0}^{t_N} u^T(\sigma) u(\sigma) d\sigma \right) \quad (3.49)$$

subject to the constraints given by (3.3). Then the *ordinary cross validation cost*

$V_0(\lambda)$ is defined as

$$V_0(\lambda) = \frac{1}{N+1} \sum_{k=0}^N \|r_k - Cx_\lambda^k(t_k)\|^2. \quad (3.50)$$

Finally we pick up an optimal value of the regularization parameter as

$$\lambda^* = \underset{\lambda > 0}{\operatorname{argmin}} (V_0(\lambda)). \quad (3.51)$$

For the problem under consideration, the special structure of the underlying dynamical system yields a nice form for the ordinary cross validation cost.

Now we solve the optimization problem (3.49) by following the path described in Section 3.3. By following the co-state approach we can conclude that the optimal trajectory will be a base integral curve of the associated Hamiltonian vector field, with suitable jump conditions on the co-state variables. It can be easily observed that the co-state variables are continuous at the left-out point, without any jump. Then with a little bit of algebra we can show that $x_\lambda^k(0)$, an optimal initial condition,

will satisfy a modified form of (3.48), in particular

$$\begin{aligned}
& [0 \ \mathbb{I}_n] \left(\prod_{i=k}^{N-1} M\Upsilon_i \right) \Upsilon_{k-1} \left(\prod_{i=0}^{k-2} M\Upsilon_i \right) \begin{bmatrix} \mathbb{I}_n \\ -\frac{1}{\lambda} C^T C \end{bmatrix} x_\lambda^k(0) \\
&= -[0 \ \mathbb{I}_n] \sum_{i=0}^{k-1} \left(\prod_{j=k}^{N-1} M\Upsilon_j \right) \Upsilon_{k-1} \left(\prod_{j=i}^{k-2} M\Upsilon_j \right) \Gamma r_i \\
&\quad - [0 \ \mathbb{I}_n] \sum_{i=k+1}^N \left(\prod_{j=i}^{N-1} M\Upsilon_j \right) \Gamma r_i
\end{aligned} \tag{3.52}$$

where Υ_i 's are obtained by factorization of Λ_i 's, as mentioned in *Lemma 4.1*. Therefore the reconstruction error encountered at the k -th data point can be represented as

$$\begin{aligned}
& r_k - C x_\lambda^k(t_k) \\
&= r_k - C \begin{bmatrix} \mathbb{I}_n \\ 0 \end{bmatrix}^T \Upsilon_{k-1} \left(\prod_{i=0}^{k-2} M\Upsilon_i \right) \begin{bmatrix} \mathbb{I}_n \\ -\frac{1}{\lambda} C^T C \end{bmatrix} x_\lambda^k(0) + \sum_{i=0}^{k-1} \left(\prod_{j=i}^{k-2} M\Upsilon_j \right) \Gamma r_i
\end{aligned} \tag{3.53}$$

when we start the trajectory from $x_\lambda^k(t_0)$ and apply the optimal input u_λ^k . From (3.52) it is quite clear that λ affects $x_\lambda^k(t_0)$ through M , Γ and $[\mathbb{I}_n \ -\frac{1}{\lambda} C^T C]$, and hence the reconstruction error is a vector of rational functions in λ . Now we can represent the cross validation cost, $V_0(\lambda)$, associated with this particular problem as

$$V_0(\lambda) = \frac{1}{N+1} \sum_{k=0}^N \left(r_k^T r_k + (x_\lambda^k(t_k))^T C^T C x_\lambda^k(t_k) - 2r_k^T C x_\lambda^k(t_k) \right). \tag{3.54}$$

As we have the (somewhat) closed form for the OCV cost, given by (3.54), we are now ready to write down the first order necessary condition for the optimality of the regularization parameter λ . We can easily check that the optimal value, λ^* , will satisfy the following first order condition

$$\sum_{k=0}^N \left((Cx_{\lambda^*}^k(t_k) - r_k)^T C \begin{bmatrix} \mathbb{I}_n \\ 0 \end{bmatrix}^T \Upsilon_{k-1} \right) \times \left(\frac{\partial}{\partial \lambda} \begin{bmatrix} \prod_{i=0}^{k-2} M\Upsilon_i \\ -\frac{1}{\lambda} C^T C \end{bmatrix} x_{\lambda}^k(0) + \sum_{i=0}^{k-1} \left(\prod_{j=i}^{k-2} M\Upsilon_j \right) \Gamma r_i \right)_{\lambda^*} = 0. \quad (3.55)$$

Remark 3.9. *The optimal value of the regularization parameter (λ^*) depends on the signal-to-noise ratio (higher SNR will cause a lower value for λ^*). In many practical applications SNR might not be constant during data capture (in case of position measurement using multi-camera network this might occur because of poor calibration around the edges of the capture volume), and in that case using the same λ will result in an erroneous reconstruction. However our algorithm works fine for piecewise constant values of λ , with transitions occurring at the sampling instances $\{t_i\}_{i=1}^N$ (this can be shown rigorously with a little modification in the application of path independence lemmas). In that case the optimal values for λ (for different segments of the trajectory) can be computed by minimizing the OCV cost over a grid of possible λ -values (each dimension signifying a particular segment of the flight).*

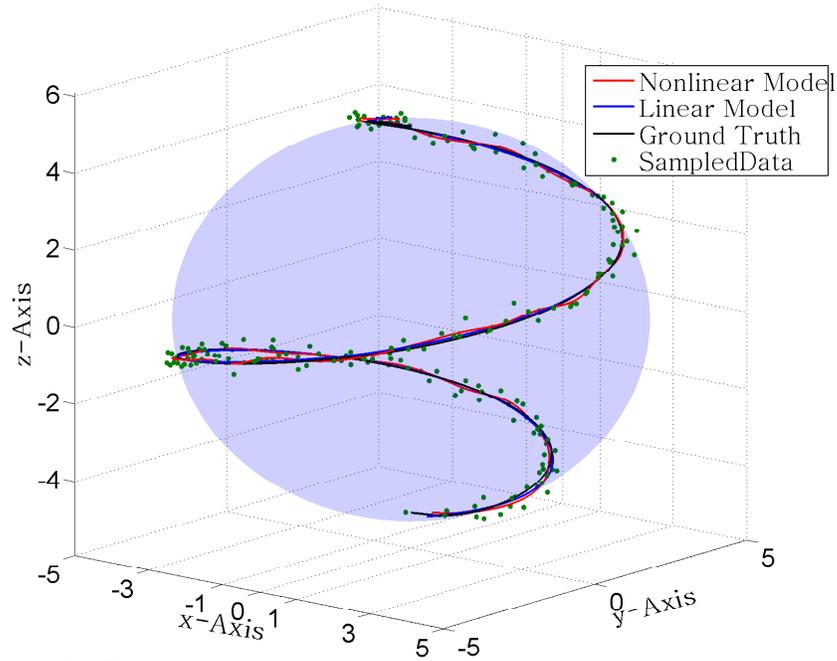
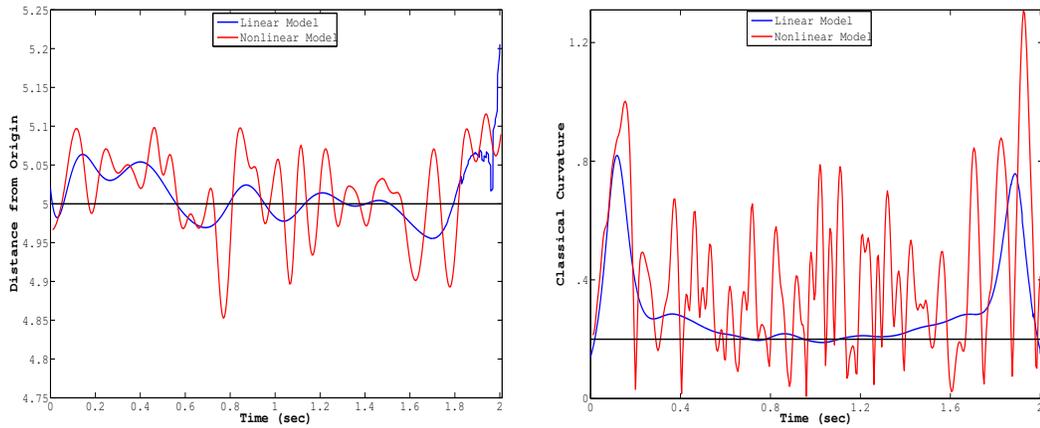


Figure 6: This figure illustrates the reconstruction of a curve on a sphere.



(a) This subplot shows the distance of the reconstructed curves from the center of the underlying sphere. Jerk path integral minimization based approach yields a curve which is closer to the sphere (in an average sense). (b) This subplot shows the classical curvature of the reconstructed curves. The curve, obtained by minimization of jerk path integral, has a curvature value greater than (almost always) the theoretical limit, i.e. 0.2.

Figure 7: The proposed algorithm, when applied to reconstruct a spiral on a sphere, performs better than the nonlinear optimization based algorithm.

3.5 Numerical Results

First we test our algorithm on a synthetic data set; the sampled trajectory data is obtained by adding an i.i.d. Gaussian noise (mean = 0, and standard deviation = 0.15) to a spiral on a sphere of radius 5, and the number of samples used was $N = 201$. The reconstruction (Fig 6) through nonlinear optimization yields an average fit error of 0.0599, whereas the proposed approach (3.5) yields an average fit error of 0.0684. However, we compute two other metrics for performance comparison, namely the distance of the curve from the center of the sphere (Fig 7a) and the curvature of the reconstructed trajectory (Fig 7b). By analyzing these two metrics it can be concluded that the proposed approach does a better job in trajectory reconstruction.

Next, with permission from Kaushik Ghose and Cynthia Moss at the Auditory Neuroethology Laboratory (BATLAB), Department of Psychology, University of Maryland, we apply our algorithm to reconstruct a bat-insect trajectory pair. The trajectory data was collected by Kaushik Ghose, and has previously been reported in the context of prey capture flight strategies by echolocating bats [Ghose *et al.*, 2006]. The particular event, that we consider, had a flight duration of around 1.93s, and the corresponding data capture rate was set at 240fps. In this case, the average fit error for the reconstruction of bat trajectory (Fig 8) through nonlinear optimization is 2.2401×10^{-4} , whereas the proposed approach yields a smaller error of 7.6142×10^{-5} . Similarly, for the insect trajectory, our approach gives a better fit (2.3913×10^{-5} compared to 1.2966×10^{-4}). From Fig 10a and Fig 10b we can

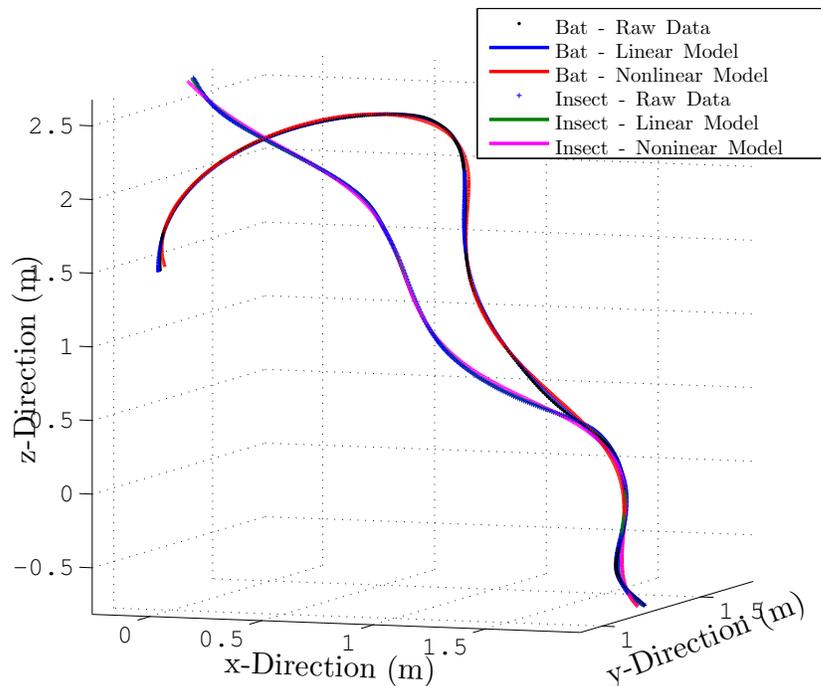
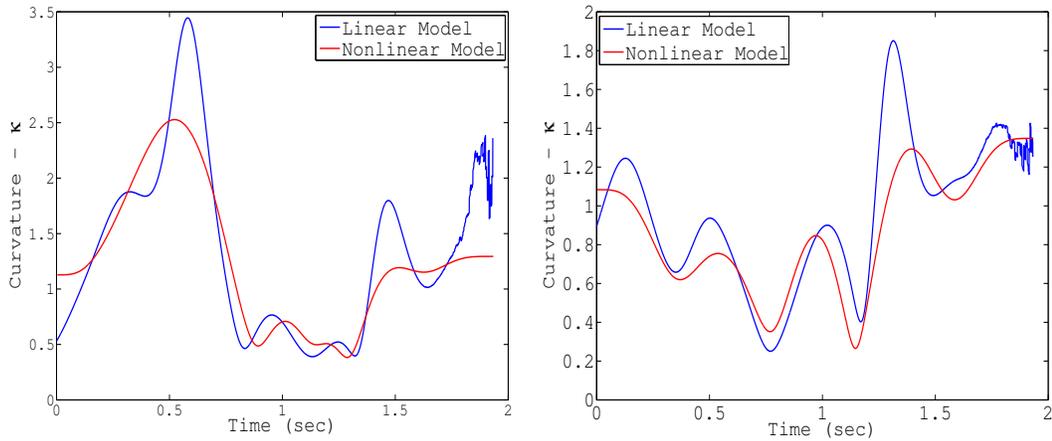


Figure 8: *This figure illustrates the reconstruction of a bat-insect trajectory pair.*

notice that the reconstructed speeds from two different regularization approaches are almost equal for both trajectories, while the same does not hold true for the evolution of curvature (Fig 9a and Fig 9b).

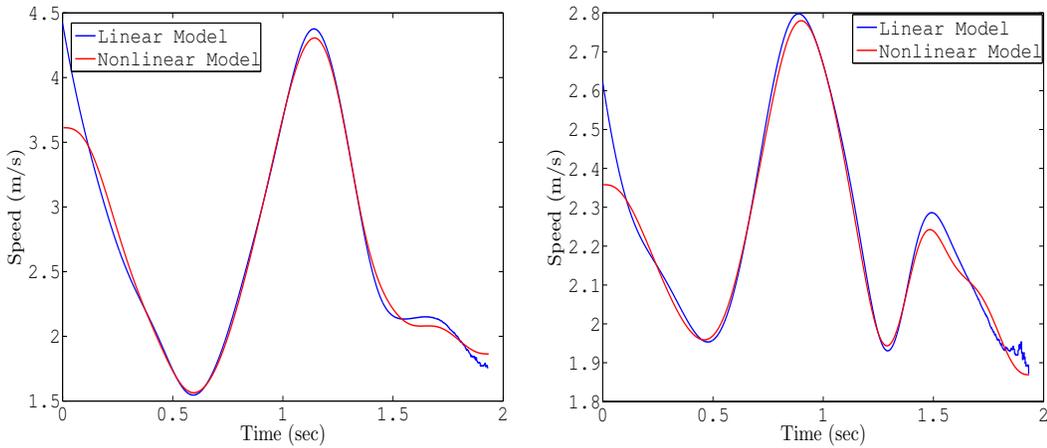
3.6 Conclusion

Using a simple linear generative model for trajectories (a triple integrator, with *jerk* as the control input), we have developed a tool to obtain analytic solutions to the inverse problem of trajectory reconstruction. Our approach casts the problem in a linear framework with quadratic cost, and solve it by using techniques from linear quadratic optimal control theory. Moreover, it has been shown that the



(a) It shows the evolution of curvature(κ) for the bat trajectory.. (b) It shows the evolution of curvature(κ) for the insect trajectory.

Figure 9: There is noticeable difference in the curvature profile obtained from two different approaches.



(a) It shows the evolution of speed(v) for the bat trajectory. (b) It shows the evolution of speed(v) for the insect trajectory.

Figure 10: The reconstructed speeds from two different approaches are almost the same.

reconstructed positions can be expressed as a linear combination of measured data. After trying the algorithm on synthetic data, we have applied it to reconstruct flight trajectories of European starling flocks (as discussed in Chapter 6). Although this approach overcomes the issues associated with the numerical optimization based

technique, it should be noted here that *not all problem can be cast in a linear quadratic framework*, and that provides motivation for the next chapter.

Chapter 4: Data Smoothing through Nonlinear Optimization - Maximum Principle

As discussed earlier, the problem of reconstructing an underlying smooth signal from sampled noisy observations arises in many areas of science and engineering (*e.g.* trajectory reconstruction [Dey & Krishnaprasad, 2012], control theoretic splines [Egerstedt & Martin, 2010] and quantum splines [Brody *et al.*, 2012]), and in many cases the underlying generative model, along with the regularizing penalty term, do not allow us to cast the data smoothing problem in a linear quadratic framework. One such example is the nonlinear version of the trajectory reconstruction problem [(2.1), discussed in Section 2.1]. A quick reference to (3.6) reveals that this problem can never be cast in a linear quadratic framework, and one needs appropriate tools for data smoothing in nonlinear settings. However, the previous results towards this direction are restricted to problems in a Riemannian setting (Burnett *et al.* [2013]; Crouch & Leite [1991, 1995]; Jakubiak *et al.* [2006]; Machado *et al.* [2010]). Although these works use calculus of variation based techniques, our earlier works on data smoothing in a linear quadratic framework (in particular, Section 3.3) has provided some insight about the applicability of Pontryagin's maximum principle [Pontryagin *et al.*, 1962; Sussmann & Willems, 1997] in a sub-Riemannian setting.

This chapter¹ focuses its attention on the nonlinear aspects of data smoothing, and develops a general framework to address data smoothing problems from a control theoretic perspective. Section 4.1 of this chapter presents a modified version of the maximum principle to solve data smoothing problems on \mathbb{R}^n , and this modified maximum principle introduces jump discontinuities in the costate variables. Later, in Section 4.2 we extend our result to address data smoothing problems in finite dimensional Lie group settings. This framework, being a modified (extended) version of Pontryagin's maximum principle, can easily be exploited to solve problems in sub-Riemannian setting as well (as demonstrated in Sections 4.4 and 4.5).

4.1 Data smoothing in a Euclidean setting

In this section we propose the modified maximum principle to address data smoothing problems in a Euclidean setting. By introducing a nonlinear generative model as

$$\dot{q}(t) = f(t, q(t), u(t)), \quad (4.1)$$

¹A significant portion of this chapter has been reproduced from a paper by Dey & Krishnaprasad [2014a].

the data smoothing problem can be formulated as the following optimal control problem on \mathbb{R}^n :

$$\begin{aligned} \text{Minimize}_{q(t_0), u} \quad & J(q(t_0), u) = \int_{t_0}^{t_N} L(t, q(t), u(t)) dt + \sum_{i=0}^N F(q(t_i), q_i) \\ \text{subject to} \quad & \text{System dynamics (4.1),} \end{aligned} \tag{4.2}$$

$$x(t_0) \in \mathbb{R}^n, \quad u \in \mathcal{U},$$

where \mathcal{U} is the space of piecewise continuous functions defined on the interval $[t_0, t_N]$. Moreover, $F(q(t_i), q_i)$ denotes the fit-error incurred at the i -th data point q_i (at time t_i), and the Lagrangian $L(t, q, u)$ introduces regularization into the data smoothing problem.

Before going into the details of necessary conditions for a sub-Riemannian optimal control problem, we focus on a special case, namely the Riemannian dynamics given by $f(t, q, u) = u$. Clearly, for this special case, (4.2) can be perceived as a calculus of variation problem. Clearly, first variation of the cost can be expressed as

$$\delta J = \int_{t_0}^{t_N} \left[\left(\frac{\partial L}{\partial q} \right)^T \delta q + \left(\frac{\partial L}{\partial \dot{q}} \right)^T \delta \dot{q} \right] dt + \sum_{i=0}^N \left(\frac{\partial F}{\partial q(t_i)} \right)^T \delta q(t_i). \tag{4.3}$$

By summing up the identity (from exact differential)

$$\int_{t_i^+}^{t_{i+1}^-} \left[\left(\frac{\partial L}{\partial \dot{q}} \right)^T \delta \dot{q} + \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right)^T \delta q \right] dt = \left[\frac{\partial L}{\partial \dot{q}}(t_{i+1}^-) \right]^T \delta q(t_{i+1}) - \left[\frac{\partial L}{\partial \dot{q}}(t_i^+) \right]^T \delta q(t_i)$$

over the intervals $(t_0^+, t_1^-), (t_1^+, t_2^-), \dots, (t_{N-1}^+, t_N^-)$, we obtain

$$\begin{aligned} \int_{t_0}^{t_N} \left(\frac{\partial L}{\partial \dot{q}} \right)^T \delta \dot{q} dt &= \left[\frac{\partial L}{\partial \dot{q}}(t_N^-) \right]^T \delta q(t_N) + \sum_{i=1}^{N-1} \left[\frac{\partial L}{\partial \dot{q}}(t_i^-) - \frac{\partial L}{\partial \dot{q}}(t_i^+) \right]^T \delta q(t_i) \\ &\quad - \left[\frac{\partial L}{\partial \dot{q}}(t_0^+) \right]^T \delta q(t_0) - \int_{t_0}^{t_N} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right)^T \delta q dt. \end{aligned} \quad (4.4)$$

Then, by using the relationship given by (4.4), the first variation (4.3) can be expressed as

$$\begin{aligned} \delta J &= \int_{t_0}^{t_N} \left[\frac{\partial L}{\partial q} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) \right]^T \delta q dt + \left[\frac{\partial L}{\partial \dot{q}}(t_N^+) \right]^T \delta q(t_N) \\ &\quad + \sum_{i=0}^N \left[\frac{\partial L}{\partial \dot{q}}(t_i^-) - \frac{\partial L}{\partial \dot{q}}(t_i^+) + \frac{\partial F}{\partial q(t_i)} \right]^T \delta q(t_i) - \left[\frac{\partial L}{\partial \dot{q}}(t_0^-) \right]^T \delta q(t_0). \end{aligned} \quad (4.5)$$

Therefore, first order necessary conditions ($\delta J = 0$) for minimality (in this special Riemannian case, $f(t, q, u) = u$) can be expressed as

$$\begin{aligned} \text{(EL)} \quad & \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = 0, \quad t \in (t_i, t_{i+1}), \quad i = 0, 1, \dots, N-1 \\ \text{(JC-1)} \quad & \frac{\partial L}{\partial \dot{q}}(t_i^+) - \frac{\partial L}{\partial \dot{q}}(t_i^-) = \frac{\partial F}{\partial q(t_i)}, \quad i = 0, 1, \dots, N \\ \text{(BC-1)} \quad & \frac{\partial L}{\partial \dot{q}}(t_0^-) = 0, \quad \text{and} \quad \frac{\partial L}{\partial \dot{q}}(t_N^+) = 0. \end{aligned} \quad (4.6)$$

Now we consider second order necessary conditions for optimality. From (4.3)

we can note that the second variation of the cost can be represented as

$$\begin{aligned} \delta^2 J &= \int_{t_0}^{t_N} \left[(\delta q)^T \left(\frac{\partial^2 L}{\partial q \partial q} \right) (\delta q) + 2(\delta q)^T \left(\frac{\partial^2 L}{\partial q \partial \dot{q}} \right) (\delta \dot{q}) + (\delta \dot{q})^T \left(\frac{\partial^2 L}{\partial \dot{q} \partial \dot{q}} \right) (\delta \dot{q}) \right] dt \\ &\quad + \sum_{i=0}^N (\delta q(t_i))^T \left(\frac{\partial^2 F}{\partial q(t_i) \partial q(t_i)} \right) (\delta q(t_i)), \end{aligned} \quad (4.7)$$

and by summing up the identity (from exact differential)

$$\begin{aligned} &\int_{t_i^+}^{t_{i+1}^-} \left[2(\delta q)^T \left(\frac{\partial^2 L}{\partial q \partial \dot{q}} \right) (\delta \dot{q}) + (\delta q)^T \left(\frac{d}{dt} \cdot \frac{\partial^2 L}{\partial q \partial \dot{q}} \right) (\delta q) \right] dt \\ &= (\delta q(t_{i+1}))^T \left[\frac{\partial^2 L}{\partial q \partial \dot{q}}(t_{i+1}^-) \right] (\delta q(t_{i+1})) - (\delta q(t_i))^T \left[\frac{\partial^2 L}{\partial q \partial \dot{q}}(t_i^+) \right] (\delta q(t_i)) \end{aligned}$$

over the intervals $(t_0^+, t_1^-), (t_1^+, t_2^-), \dots, (t_{N-1}^+, t_N^-)$, we obtain

$$\begin{aligned} &\int_{t_0}^{t_N} 2(\delta q)^T \left(\frac{\partial^2 L}{\partial q \partial \dot{q}} \right) (\delta \dot{q}) dt \\ &= (\delta q(t_N))^T \left[\frac{\partial^2 L}{\partial q \partial \dot{q}}(t_N^-) \right] (\delta q(t_N)) - (\delta q(t_0))^T \left[\frac{\partial^2 L}{\partial q \partial \dot{q}}(t_0^+) \right] (\delta q(t_0)) \\ &\quad + \sum_{i=1}^{N-1} (\delta q(t_i))^T \left[\frac{\partial^2 L}{\partial q \partial \dot{q}}(t_i^-) - \frac{\partial^2 L}{\partial q \partial \dot{q}}(t_i^+) \right] (\delta q(t_i)) \\ &\quad - \int_{t_0}^{t_N} (\delta q)^T \left(\frac{d}{dt} \cdot \frac{\partial^2 L}{\partial q \partial \dot{q}} \right) (\delta q) dt. \end{aligned} \quad (4.8)$$

Hence, the second variation (4.7) can be expressed as

$$\begin{aligned}
\delta^2 J &= \int_{t_0}^{t_N} (\delta q)^T \left[\frac{\partial^2 L}{\partial q \partial q} - \frac{d}{dt} \cdot \frac{\partial^2 L}{\partial q \partial \dot{q}} \right] (\delta q) dt + \int_{t_0}^{t_N} (\delta \dot{q})^T \left[\frac{\partial^2 L}{\partial \dot{q} \partial \dot{q}} \right] (\delta \dot{q}) dt \\
&+ \sum_{i=0}^N (\delta q(t_i))^T \left[\frac{\partial^2 L}{\partial q \partial \dot{q}}(t_i^-) - \frac{\partial^2 L}{\partial q \partial \dot{q}}(t_i^+) + \frac{\partial^2 F}{\partial q(t_i) \partial q(t_i)} \right] (\delta q(t_i)) \\
&+ (\delta q(t_N))^T \left[\frac{\partial^2 L}{\partial q \partial \dot{q}}(t_N^+) \right] (\delta q(t_N)) - (\delta q(t_0))^T \left[\frac{\partial^2 L}{\partial q \partial \dot{q}}(t_0^-) \right] (\delta q(t_0)). \quad (4.9)
\end{aligned}$$

This leads us to express second order necessary conditions ($\delta^2 J \geq 0$) as

$$\begin{aligned}
\text{(LE)} \quad & \frac{\partial^2 L}{\partial \dot{q} \partial \dot{q}} \geq 0, \quad t \in (t_i, t_{i+1}), \quad i = 0, 1, \dots, N-1 \\
\text{(JC-2)} \quad & \frac{\partial^2 L}{\partial q \partial \dot{q}}(t_i^+) - \frac{\partial^2 L}{\partial q \partial \dot{q}}(t_i^-) = \frac{\partial^2 F}{\partial q(t_i) \partial q(t_i)}, \quad i = 0, 1, \dots, N \quad (4.10) \\
\text{(BC-2)} \quad & \frac{\partial^2 L}{\partial q \partial \dot{q}}(t_0^-) = 0, \quad \text{and} \quad \frac{\partial^2 L}{\partial q \partial \dot{q}}(t_N^+) = 0.
\end{aligned}$$

At this point we introduce the adjoint/co-state variable (p) and define the pre-Hamiltonian as

$$H(t, q, p, u) = p^T u - L(t, q, u), \quad (4.11)$$

and therefore

$$\begin{aligned}
\frac{\partial H}{\partial p} &= u, & \frac{\partial H}{\partial q} &= -\frac{\partial L}{\partial q}, \\
\frac{\partial H}{\partial u} &= p - \frac{\partial L}{\partial u}, & \frac{\partial^2 H}{\partial u^2} &= -\frac{\partial^2 L}{\partial u^2}.
\end{aligned}$$

It is worth mentioning here that we are focusing on regular extremals, i.e. $\frac{\partial^2 L}{\partial \dot{q} \partial \dot{q}} \neq 0$ along any solution of the Euler-Lagrange (**EL**) equation (4.6). Now we assume that

$t \mapsto q^*(t)$ is a trajectory of the system $\dot{q} = f(t, q, u) = u$ which solves the optimal control problem (4.2) with control given as $u^*(t) = \dot{q}^*(t)$, and define

$$p(t) = \frac{\partial L}{\partial u}(t, q^*(t), \dot{q}^*(t)). \quad (4.12)$$

Then by applying the Euler-Lagrange condition (EL) from (4.6) we obtain

$$\dot{p}(t) = \frac{\partial L}{\partial q}(t, q^*(t), \dot{q}^*(t)) = -\frac{\partial H}{\partial q}(t, q^*(t), p(t), \dot{q}^*(t)), \quad (4.13)$$

and clearly we have

$$\dot{q}^*(t) = u^*(t) = \frac{\partial H}{\partial p}(t, q^*(t), p(t), \dot{q}^*(t)). \quad (4.14)$$

Also from the definition of the adjoint variable we have

$$\frac{\partial H}{\partial u}(t, q^*(t), p(t), \dot{q}^*(t)) = 0, \quad (4.15)$$

and the Legendre condition (LE) yields

$$\frac{\partial^2 H}{\partial u^2}(t, q^*(t), p(t), \dot{q}^*(t)) < 0 \quad (4.16)$$

because of the regularity property. As, (4.15) and (4.16) maximize the pre-Hamiltonian, we have

$$H(t, q^*(t), p(t), \dot{q}^*(t)) = \text{Max}_u H(t, q^*(t), p(t), u) \quad (4.17)$$

A generalization of the above results leads us to an alternative version of the Pontryagin's maximum principle, which is tailored for the data smoothing problem.

Theorem 4.1 (PMP for data smoothing). *Consider an optimal control problem on \mathbb{R}^n , given as*

$$\begin{aligned} \text{Minimize}_{q(t_0); u} \quad & J(q(t_0), u) = \int_{t_0}^{t_N} L(t, q(t), u(t)) dt + \sum_{i=0}^N F(q(t_i), q_i) \\ \text{subject to:} \quad & \dot{q}(t) = f(t, q(t), u(t)), \\ & q : [t_0, t_N] \rightarrow \mathbb{R}^n, \\ & u \in \mathcal{U} : [t_0, t_N] \rightarrow U \subset \mathbb{R}^m, \quad u - \text{piecewise continuous.} \end{aligned} \tag{4.18}$$

Now we assume that u^* is an optimal control input for (4.18), and q^* denotes the corresponding state trajectory. Then there exists a costate trajectory $p : [t_0, t_N] \rightarrow \mathbb{R}^n$ such that

$$\begin{aligned} \dot{q}^*(t) &= \frac{\partial H}{\partial p}(t, q^*(t), p(t), u^*(t)) \\ \dot{p}(t) &= -\frac{\partial H}{\partial q}(t, q^*(t), p(t), u^*(t)) \end{aligned} \tag{4.19}$$

during $t \in (t_i, t_{i+1})$, $i = 0, 1, \dots, N$, and

$$H(t, q^*, p, u^*) = \max_{u \in U} H(t, q^*, p, u) \tag{4.20}$$

for $t \in [t_0, t_N] \setminus \{t_0, t_1, \dots, t_N\}$, where the pre-Hamiltonian is defined as $H(t, q, p, u) = \langle p, f(t, q, u) \rangle - L(t, q, u)$. Moreover, the intermediate state cost terms require jump discontinuities in the costate variables, and the jump conditions and the boundary

values are given as

$$\begin{aligned}
p(t_0^-) &= 0, \\
p(t_i^+) - p(t_i^-) &= \frac{\partial F(q(t_i), q_i)}{\partial q(t_i)}, \quad i = 0, 1, \dots, N, \\
p(t_N^+) &= 0.
\end{aligned} \tag{4.21}$$

Remark 4.1. *A quick revisit to the trajectory reconstruction problem [(3.5), discussed in Section 3.1] illustrates the control theoretic formulation (4.18) of a data smoothing problem on \mathbb{R}^n . It is easy to verify that this trajectory reconstruction problem can be treated as a special case of (4.18) by introducing the following correspondences*

$$\text{Lagrangian:} \quad L = \lambda u^T u$$

$$\text{Generative Model:} \quad f(t, q(t), u(t)) = Aq(t) + Bu(t), \quad q(t) \in \mathbb{R}^9, \quad u(t) \in \mathbb{R}^3$$

$$\begin{aligned}
\text{Fit Error:} \quad F(q(t_i), r_i) &= \|Cq(t_i) - r_i\|^2 \\
&= q^T(t_i)C^T Cq(t_i) - 2x^T(t_i)C^T r_i + r_i^T r_i,
\end{aligned}$$

where r_i denotes the measured position at time t_i and $C \in \mathbb{R}^{3 \times 9}$ maps the states into the outputs.

Remark 4.2. *It is worth mentioning here that the fit cost enters the problem through the jump conditions in the co-state variables; while the flow of the system between two consecutive data points is dictated by the path cost (penalty term) only.*

Now we venture into the detailed proof of Theorem 4.1, and by adopting an approach similar to the one taken in the book by Liberzon [2011], we develop the proof using needle (strong) variation.

Proof. At the outset we introduce a new state variable $\tilde{q} : [t_0, t_N] \rightarrow \mathbb{R}$ with its dynamics governed by

$$\begin{aligned} \dot{\tilde{q}}(t) &= L(t, q(t), u(t)), & t \in (t_i, t_{i+1}), \\ \tilde{q}(t_i^+) - \tilde{q}(t_i^-) &= F(q(t_i), q_i), & i = 0, 1, \dots, N, \\ \tilde{q}(t_0^-) &= 0, \end{aligned} \tag{4.22}$$

and this leads us to an augmented system. By introducing

$$y(t) \triangleq \begin{pmatrix} \tilde{q}(t) \\ q(t) \end{pmatrix} \in \mathbb{R}^{n+1}, \tag{4.23}$$

the dynamics of this augmented system (4.23) can be represented as

$$\dot{y}(t) = \begin{pmatrix} L(t, q(t), u(t)) \\ f(t, q(t), u(t)) \end{pmatrix} \triangleq g(t, y(t), u(t)), \tag{4.24}$$

with the initial condition $y(t_0^-) = \begin{pmatrix} 0 \\ q(t_0) \end{pmatrix}$. The corresponding jump conditions

in y can be expressed as

$$y(t_i^+) - y(t_i^-) = \begin{pmatrix} F(q(t_i), q_i) \\ 0 \end{pmatrix}.$$

As a consequence, the cost functional (in 4.18) can be expressed as

$$J(q(t_0), u) = \tilde{q}(t_N^+) = J(y(t_0), u). \quad (4.25)$$

Clearly, an optimal trajectory q^* (generated by u^*) of the original system (4.18) can be retrieved from an optimal trajectory y^* of the augmented system (4.23) through a projection onto \mathbb{R}^n along the \tilde{q} -axis.

Now, let $a > 0$, $b \in \mathbb{R}$ be such that $I_\epsilon = (b - \epsilon a, b] \subset (t_0, t_N)$, $t_i \notin (b - a, b] \forall i$, and u^* is continuous on I_ϵ , $\forall \epsilon \in (0, 1]$ and at b . Next we introduce a needle variation (a perturbation pulse of short duration) by defining the perturbed control as

$$u_{w, I_\epsilon}(t) \triangleq \begin{cases} u^*(t) & \text{if } t \notin I_\epsilon \\ w & \text{if } t \in I_\epsilon \end{cases}, \quad (4.26)$$

where $w \in U$. Then, by letting k denote the index such that $I_\epsilon \subset (t_{k-1}, t_k)$, we have

$$y(t) = y^*(t) + \epsilon \Phi_*(t, b) \delta(w, a, b) + O(\epsilon^2) \quad (4.27)$$

for $b \leq t \leq t_k^-$ [Liberzon, 2011, Section 4.2.4]. Here, the perturbation term $\delta(w, a, b)$

is defined as

$$\delta(w, a, b) = a \left(g(b, y^*(b), w) - g(b, y^*(b), u^*(b)) \right), \quad (4.28)$$

and Φ_* denotes the state transition matrix for the linearized dynamics governed by

$$\dot{\psi}(t) = \begin{bmatrix} 0 & (L_q(t, q^*(t), u^*(t)))^T \\ 0_{n \times 1} & f_q(t, q^*(t), u^*(t)) \end{bmatrix} \psi(t).$$

Now we introduce a matrix Γ_k , defined as

$$\Gamma_k = \begin{bmatrix} 1 & \left(\frac{\partial F}{\partial q}(q^*(t_k), q_k) \right)^T \\ 0_{n \times 1} & I_{n \times n} \end{bmatrix},$$

to capture the effect of jump discontinuities on the perturbed trajectory y , where

$\frac{\partial F}{\partial q}$ denotes the partial derivative of F with respect to its first argument. This

enables us to express $y(t_k^+)$ as

$$y(t_k^+) = y^*(t_k^+) + \epsilon \Gamma_k \Phi_*(t_k, b) \delta(w, a, b) + O(\epsilon^2). \quad (4.29)$$

Proceeding this way the terminal point of the perturbed trajectory can be expressed

as

$$y(t_N^+) = y^*(t_N^+) + \epsilon \Gamma_N \Phi_*(t_N, t_{N-1}) \cdots \Gamma_k \Phi_*(t_k, b) \delta(w, a, b) + O(\epsilon^2). \quad (4.30)$$

Now, $\epsilon \Gamma_N \Phi_*(t_N, t_{N-1}) \cdots \Gamma_k \Phi_*(t_k, b) \delta(w, a, b)$ can be interpreted as an infinites-

imal perturbation of the terminal state caused by the needle variation in the control input (u_w, I_ϵ) , and its direction depends only on b and w (4.28,4.30). By letting $\rho(w, b)$ denote the ray in this direction originating at $y^*(t_N^+)$, we define \vec{P} as the union of the rays $\rho(w, b)$ for all possible values of w and b . It can be noticed that the cone \vec{P} is not convex in general, and hence we concatenate different needle variations to generate a larger cone with the same vertex. These concatenations yield a larger cone which contains the convex combinations of the points in \vec{P} [Liberzon, 2011, Section 4.2.5], and we call it the terminal cone ($TC(y^*(t_N^+))$). Therefore, there exists a nonzero vector $\mu \in \mathbb{R}^{n+1}$ such that

$$\mu^T (y(t_N^+) - y^*(t_N^+)) \geq 0 \quad (4.31)$$

for any perturbed trajectory y such that $y(t_N^+) - y^*(t_N^+) \in TC(y^*(t_N^+))$ [Tits, 2013, Theorem B.3]. Then, by following the arguments given in [Liberzon, 2011, Section 4.2.6], we can claim that (4.31) must be satisfied for the choice of

$$\mu^* = \begin{bmatrix} 1 \\ 0_{n \times 1} \end{bmatrix}, \quad (4.32)$$

because otherwise there exists a $y(t_N^+)$ which would violate the optimality of y^* . By using this choice of $\mu = \mu^*$ for the perturbed trajectories of the form (4.30), (4.31) can be represented as

$$[\Phi_*^T(t_k, b)\Gamma_k^T \cdots \Phi_*^T(t_N, t_{N-1})\Gamma_N^T \mu^*]^T \delta(w, a, b) \geq 0. \quad (4.33)$$

Now we introduce $\xi : [t_0, t_N] \mapsto \mathbb{R}^{n+1}$, and by letting the dynamics of ξ be governed by

$$\dot{\xi}(t) = \begin{bmatrix} 0 & 0_{n \times 1} \\ -L_q(t, q^*(t), u^*(t)) & -(f_q(t, q^*(t), u^*(t)))^T \end{bmatrix} \xi(t), \quad (4.34)$$

we define an adjoint system on \mathbb{R}^{n+1} . Moreover, we define $\xi(t_N^+) = \mu^*$, and introduce the interface conditions given by

$$\xi(t_i^-) = \begin{bmatrix} 1 & 0_{1 \times n} \\ \frac{\partial F}{\partial q}(q^*(t_k), q_k) & I_{n \times n} \end{bmatrix} \xi(t_i^+) \quad (4.35)$$

at t_N, t_{N-1}, \dots, t_0 . This adjoint system allows us to represent (4.33) as

$$\xi^T(b) \left(g(b, y^*(b), w) - g(b, y^*(b), u^*(b)) \right) \geq 0 \quad (4.36)$$

for all $w \in U$ and $b \in [t_0, t_N] \setminus \{t_0, \dots, t_N\}$.

It can be easily verified that the first component of $\dot{\xi}$ is identically zero (4.34), and the jump discontinuity doesn't exist for the first component of ξ (4.35). Then it directly follows from (4.32) that the first component of ξ is set constant at 1. This enables us to decompose $\xi(t)$ into

$$\xi(t) = \begin{bmatrix} 1 \\ -p(t) \end{bmatrix}. \quad (4.37)$$

This decomposition of ξ yields the dynamics, boundary values and jump conditions for $p : [t_0, t_N] \mapsto \mathbb{R}^n$ as:

$$\dot{p}(t) = -L_q(t, q^*(t), u^*(t)) + (f_q(t, q^*(t), u^*(t)))^T p(t) \quad (4.38)$$

and

$$\begin{aligned} p(t_N^+) &= 0 \\ p(t_i^+) - p(t_i^-) &= \frac{\partial F}{\partial q}(q^*(t_k), q_k). \end{aligned} \quad (4.39)$$

Using the fact that the pre-Hamiltonian is defined as

$$H(t, q, p, u) = \langle p, f(t, q, u) \rangle - L(t, q, u), \quad (4.40)$$

the dynamics of p (4.38) can be expressed as

$$\dot{p}(t) = -\frac{\partial H}{\partial q}(t, q^*, p, u^*). \quad (4.41)$$

Moreover, by using the fact that $q(t_0)$ is free and minimizes the cost, it can be concluded that $p(t_0^-) = 0$ [Liberzon, 2011, Section 4.3.1]. Also, the dynamics of the optimal state trajectory q^* can be derived straightforward from (4.40). Now we focus to (4.36) to show maximality of the Hamiltonian. By using the decomposition of ξ , (4.36) can be represented as

$$\begin{bmatrix} 1 \\ -p(t) \end{bmatrix}^T \begin{bmatrix} L(t, q^*(t), w) - L(t, q^*(t), u^*(t)) \\ f(t, q^*(t), w) - f(t, q^*(t), u^*(t)) \end{bmatrix} \geq 0,$$

or equivalently

$$H(t, q^*(t), p(t), u^*(t)) \geq H(t, q^*(t), p(t), w) \quad (4.42)$$

for $t \in [t_0, t_N] \setminus \{t_0, \dots, t_N\}$. This concludes our proof for Theorem 4.1. \square

Remark 4.3. *The maximum principle, when applied to the trajectory reconstruction problem (referred in Remark 4.1), gives rise to the Hamiltonian dynamics given by*

$$\frac{d}{dt} \begin{bmatrix} q^*(t) \\ p(t) \end{bmatrix} = \begin{bmatrix} A & \frac{1}{2\lambda} BB^T \\ 0 & -A^T \end{bmatrix} \begin{bmatrix} q^*(t) \\ p(t) \end{bmatrix},$$

and the corresponding optimal control can be expressed as

$$u^*(t) = \frac{1}{2\lambda} B^T p(t).$$

Moreover, the boundary values and jump conditions for the costate p are computed as

$$\begin{aligned} p(t_0^-) &= p(t_N^+) = 0 \\ p(t_i^+) - p(t_i^-) &= 2C^T [Cq(t_i) - r_i]. \end{aligned}$$

Although it appears slightly different from the solutions obtained in (3.38,3.39), this discrepancy can be avoided by introducing the following change of variable

$$\tilde{p} = -\frac{1}{2\lambda} p.$$

Remark 4.4. *It is worth mentioning here that jump discontinuities in optimal control problems are not new. In the context of calculus of variation, they can be traced back to the time of Weierstrass (Weierstrass-Erdmann corner condition [Liberzon, 2011]). More recently, they arise in hybrid optimal control when the associated dynamics undergo switching (Shaikh & Caines [2007]; Taringoo & Caines [2013]).*

4.2 Data smoothing problems in a Finite Dimensional Matrix Lie group setting

In this section we extend our result to tackle data smoothing problems in finite dimensional matrix Lie group settings. For example, by using natural-Frenet frame equations [Bishop, 1975] as the underlying generative model and penalizing high rates of change in speed and curvatures, trajectory reconstruction can be formulated as a data smoothing problem on $SE(3) \times \mathbb{R}^3$ [Reddy, 2007]. Also, trajectory smoothing on a sphere can be posed as a data smoothing problem on $SO(3)$ (similar to the problem discussed by Brody *et al.* [2012]).

We begin by considering a finite dimensional matrix Lie group G and a left invariant vector field defined on G . Before defining left invariance we introduce left action by letting $L_g : G \rightarrow G, h \mapsto gh$ denote the left translation by $g \in G$ and $T_h L_g : T_h G \rightarrow T_{gh} G$ denote its tangent map (linearization).

Definition 4.1 (Left invariant vector field). *Given the left action L_g defined on a*

Lie group G , a vector field $X : G \rightarrow TG, h \mapsto (h, v_h)$ will be called left invariant if

$$T_h L_g(v_h) = v_{gh} \quad \forall h \in G.$$

This allows us to define a *left invariant control system* by letting v_e (referred as ξ henceforth) be a *control curve* in the Lie algebra $\mathfrak{g}(= T_e G$, tangent space to the Lie group G at the group identity element e). Then the dynamics takes the form

$$\dot{g}(t) = T_e L_{g(t)} \cdot \xi_{u(t)} \tag{4.43}$$

where each control input w defines an element ξ_w of \mathfrak{g} .

Now we consider the following optimal control problem on G :

$$\begin{aligned} \text{Minimize}_{g(t_0); u} \quad & J(g(t_0), u) = \int_{t_0}^{t_N} L(u(t)) dt + \sum_{i=0}^N F(g(t_i), g_i) \\ \text{subject to:} \quad & \dot{g} = T_e L_g \cdot \xi_u = g \xi_u, \\ & g : [t_0, t_N] \rightarrow G, \quad u \in \mathcal{U}, \end{aligned} \tag{4.44}$$

where $F(g(t_i), g_i)$ denotes the fit error at sampling instant t_i and \mathcal{U} is the space of piecewise continuous functions over $[t_0, t_N]$. The control curve is defined as $\xi_u = X_0 + \sum_{i=1}^k u_i X_k$ where $\{X_i\}_{i=1}^n$ is a basis of the Lie algebra \mathfrak{g} , $k < n$ and $X_0 \in \text{span}\{X_{k+1}, X_{k+2}, \dots, X_n\}$. Clearly, the Lagrangian is assumed to be left invariant.

We also assume that the fit cost $F(g(t_i), g_i)$ is invariant under the left action, i.e.

$$F(g(t_i), g_i) = F(L_g \cdot g(t_i), L_g \cdot g_i)$$

for any $g \in G$, and $u(t), t \in [t_0, t_N]$ takes value in $U \subset \mathbb{R}^k$.

Theorem 4.2 (PMP for data smoothing on a finite dimensional matrix Lie Group). *Consider an optimal control problem on a finite dimensional matrix Lie group G , given as*

$$\begin{aligned} \text{Minimize}_{g(t_0); u} \quad & J(g(t_0), u) = \int_{t_0}^{t_N} L(u(t)) dt + \sum_{i=0}^N F(g(t_i), g_i) \\ \text{subject to:} \quad & \dot{g}(t) = T_e L_{g(t)} \cdot \xi_{u(t)} = g(t) \left(X_0 + \sum_{i=1}^k u_i(t) X_k \right), \end{aligned} \quad (4.45)$$

$$g : [t_0, t_N] \rightarrow G,$$

$$u \in \mathcal{U} : [t_0, t_N] \rightarrow U \subset \mathbb{R}^k, \quad u - \text{piecewise continuous.}$$

Now we assume that u^* is an optimal control input for (4.45). Then, corresponding state trajectory g^* is the base integral curve (g^*, p) of a Hamiltonian vector field $X_{H(g^*, p, u^*)}$ on T^*G , where the pre-Hamiltonian is defined as

$$H(g, p, u) = \langle p, T_e L_g \cdot \xi_u \rangle - L(u) = \langle p, \dot{g} \rangle - L(u),$$

and an optimal control input maximizes H (which clearly is G invariant), i.e.

$$H(g^*, p, u^*) = \text{Max}_u H(g^*, p, u).$$

Moreover, intermediate state penalties require jump discontinuities in p and the corresponding boundary values and jump conditions are given as

$$p(t_0^-) = p(t_N^+) = 0$$

$$\text{and, } p(t_i^+) - p(t_i^-) = D_{g^*(t_i)}F, \quad i = 0, 1, \dots, N$$

where $D_{g^*(t_i)}F$ represents the Frechet derivative of the fit-error at $g^*(t_i) \in G$.

Proof. Here we adopt a variational approach to derive necessary conditions for optimality, and as a first step express the cost functional in terms of the pre-Hamiltonian. Clearly, the cost can be represented as

$$J(g(t_0), u) = \int_{t_0}^{t_N} (\langle p, \dot{g} \rangle - H(g, p, u)) dt + \sum_{i=0}^N F(g(t_i), g_i). \quad (4.46)$$

As our focus is restricted to matrix Lie groups, the pairing $\langle \cdot, \cdot \rangle$ should be interpreted as a trace inner-product in an appropriate matrix space.

Let u^* be an optimal control (piecewise continuous) and g^* be the corresponding optimal trajectory of the system. First we consider perturbed controls of the form

$$u_\epsilon = u^* + \epsilon \delta u \quad (4.47)$$

where δu is continuous in the intervals (t_i, t_{i+1}) , and let $\xi_\epsilon = \xi_{u^*} + \epsilon \delta \xi_u$ denote the associated perturbed control curves on \mathfrak{g} . Then, the corresponding perturbed

trajectory can be represented as

$$g_\epsilon = g^* + \epsilon \delta g + O(\epsilon^2) \quad (4.48)$$

where $\delta g = g^* \delta \xi_u$.

We begin by considering the first variation of the cost functional J , and show that the first variation δJ can be expressed as

$$\begin{aligned} \delta J = & \int_{t_0}^{t_N} \left(\langle p, \delta \dot{g} \rangle - \langle \nabla_g H(g^*, p, u^*), \delta g \rangle - \langle \nabla_u H(g^*, p, u^*), \delta u \rangle \right) dt \\ & + \sum_{i=0}^N \langle D_{g^*(t_i)} F, \delta g(t_i) \rangle, \end{aligned} \quad (4.49)$$

where $D_{g^*(t_i)} F$ represents the Frechet derivative at $g^*(t_i) \in G$. Next we show that the first integrand in the first variation (4.49) can be expressed as

$$\begin{aligned} \int_{t_0}^{t_N} \langle p, \delta \dot{g} \rangle dt &= \sum_{i=0}^{N-1} \int_{t_i^+}^{t_{i+1}^-} \langle p, \delta \dot{g} \rangle dt \\ &= \sum_{i=0}^{N-1} \left(\langle p(t_{i+1}^-), \delta g(t_{i+1}^-) \rangle - \langle p(t_i^+), \delta g(t_i^+) \rangle - \int_{t_i^+}^{t_{i+1}^-} \langle \dot{p}, \delta g \rangle dt \right) \\ &= \sum_{i=0}^{N-1} \left(\langle p(t_{i+1}^-), \delta g(t_{i+1}) \rangle - \langle p(t_i^+), \delta g(t_i) \rangle \right) - \int_{t_0}^{t_N} \langle \dot{p}, \delta g \rangle dt \\ &= \langle p(t_N^+), \delta g(t_N) \rangle + \sum_{i=0}^N \langle p(t_i^-) - p(t_i^+), \delta g(t_i) \rangle - \langle p(t_0^-), \delta g(t_0) \rangle \\ &\quad - \int_{t_0}^{t_N} \langle \dot{p}, \delta g \rangle dt. \end{aligned} \quad (4.50)$$

Then by replacing (4.50) into (4.49), the first variation can be represented as

$$\begin{aligned} \delta J = & \langle p(t_N^+), \delta g(t_N) \rangle + \sum_{i=0}^N \langle p(t_i^-) - p(t_i^+) + D_{g^*(t_i)} F, \delta g(t_i) \rangle - \langle p(t_0^-), \delta g(t_0) \rangle \\ & - \int_{t_0}^{t_N} \langle \dot{p} + \nabla_g H(g^*, p, u^*), \delta g \rangle dt - \int_{t_0}^{t_N} \langle \nabla_u H(g^*, p, u^*), \delta u \rangle dt. \end{aligned} \quad (4.51)$$

Now, first-order necessary condition for optimality dictates that δJ (4.51) must be zero for any perturbation in control (δu) or initial condition ($\delta g(t_0)$), and this condition holds true for every p . Now, we make a special choice and assume the following structure on p :

$$\begin{aligned} & \bullet \quad \dot{p} = -\nabla_g H(g^*, p, u^*) \\ & \bullet \quad p(t_N^+) = 0 \\ & \bullet \quad p(t_i^+) - p(t_i^-) = D_{g^*(t_i)} F, \quad \forall i \in \{1, 2, \dots, N\}. \end{aligned} \quad (4.52)$$

These assumptions allow us to represent the first variation around an optimal trajectory as

$$\delta J = -\langle p(t_0^-), \delta g(t_0) \rangle - \int_{t_0}^{t_N} \langle \nabla_u H(g^*, p, u^*), \delta u \rangle dt, \quad (4.53)$$

and first-order necessary condition for optimality requires $\delta J = 0$ (4.53) for any perturbation δu or $\delta g(t_0)$. Thus, first-order necessary condition implies

$$\begin{aligned} & \bullet \quad \nabla_u H(g^*, p, u^*) = 0 \\ & \bullet \quad p(t_0^-) = 0. \end{aligned} \quad (4.54)$$

Moreover, the definition of the pre-Hamiltonian let us represent the dynamics of g^* as

$$\dot{g}^* = \nabla_p H(g^*, p, u^*) \tag{4.55}$$

Next we focus on the second variation of J , show that a second order necessary condition can be expressed as

$$\nabla_{uu} H(g^*, p, u^*) \leq 0. \tag{4.56}$$

Now, by narrowing our focus to regular extremals, i.e. $\nabla_{\dot{g}\dot{g}} L \neq 0$ along any solution of the (g^*, p) -dynamics, (4.54) and (4.56) yield the following maximality condition

$$H(g^*, p, u^*) = \text{Max}_u H(g^*, p, u) \tag{4.57}$$

This concludes our proof for Theorem 4.2. □

4.3 A Quick Revisit to Lie-Poisson Reduction

This section provides a brief introduction to Lie-Poisson reduction. Interested readers may refer the works of Krishnaprasad [1985, 1993] and Marsden & Ratiu [2003] for further details.

As there exists a bundle isomorphism between the cotangent bundle (T^*G) of a Lie group G and the product $G \times \mathfrak{g}^*$, we can easily introduce two bundle projections,

defined as

$$\begin{aligned}\pi : T^*G &\rightarrow \mathfrak{g}^* \\ \tilde{\pi} : T^*G &\rightarrow G.\end{aligned}\tag{4.58}$$

Moreover, any vector from a tangent space of the cotangent bundle T^*G can be translated to a vector in the Lie algebra (via consecutive action of appropriate tangent lifts, as shown in Fig 11). These two facts provide a natural choice for a one-form on T^*G , namely the *Poincaré one-form* ($T^*G \ni a \mapsto \Theta_a \in T_a^*(T^*G)$), defined as

$$\Theta_a(v) = \left\langle \pi(a), T_{\tilde{\pi}(a)}L_{\tilde{\pi}(a)^{-1}} \cdot (T_a\tilde{\pi} \cdot v) \right\rangle,\tag{4.59}$$

for any $v \in T_a(T^*G)$, $a \in T^*G$. Then, by using exterior derivative of this one-form, we can define a symplectic form on T^*G ($\omega = -d\Theta$) [Marsden & Ratiu, 2003]. Clearly, this symplectic form associates a Hamiltonian vector field to each smooth real-valued function (Hamiltonian) on T^*G .

Now, by letting $\mathfrak{C}^\infty(T^*G)$ denote the space of smooth real-valued functions on T^*G , we can introduce a Poisson bracket $\{\cdot, \cdot\} : \mathfrak{C}^\infty(T^*G) \times \mathfrak{C}^\infty(T^*G) \rightarrow \mathfrak{C}^\infty(T^*G)$, $\phi, \psi \mapsto \{\phi, \psi\} = \omega(H_\phi, H_\psi)$, where H_ϕ is the Hamiltonian vector field associated with the smooth function ϕ . It can be easily verified that if $\phi, \psi \in \mathfrak{C}^\infty(T^*G)$ are invariant under left translation, then $\{\phi, \psi\}$ is also G -invariant. As the pullback of any function on \mathfrak{g}^* by π will define a G -invariant function on T^*G , this enables us to define the Lie-Poisson bracket $\{\cdot, \cdot\}_{\mathfrak{g}^*} : \mathfrak{C}^\infty(\mathfrak{g}^*) \times \mathfrak{C}^\infty(\mathfrak{g}^*) \rightarrow \mathfrak{C}^\infty(\mathfrak{g}^*)$ as

$$\pi^*\{h_1, h_2\}_{\mathfrak{g}^*} = \{h_1, h_2\}_{\mathfrak{g}^*} \circ \pi = \{\pi^*h_1, \pi^*h_2\}\tag{4.60}$$

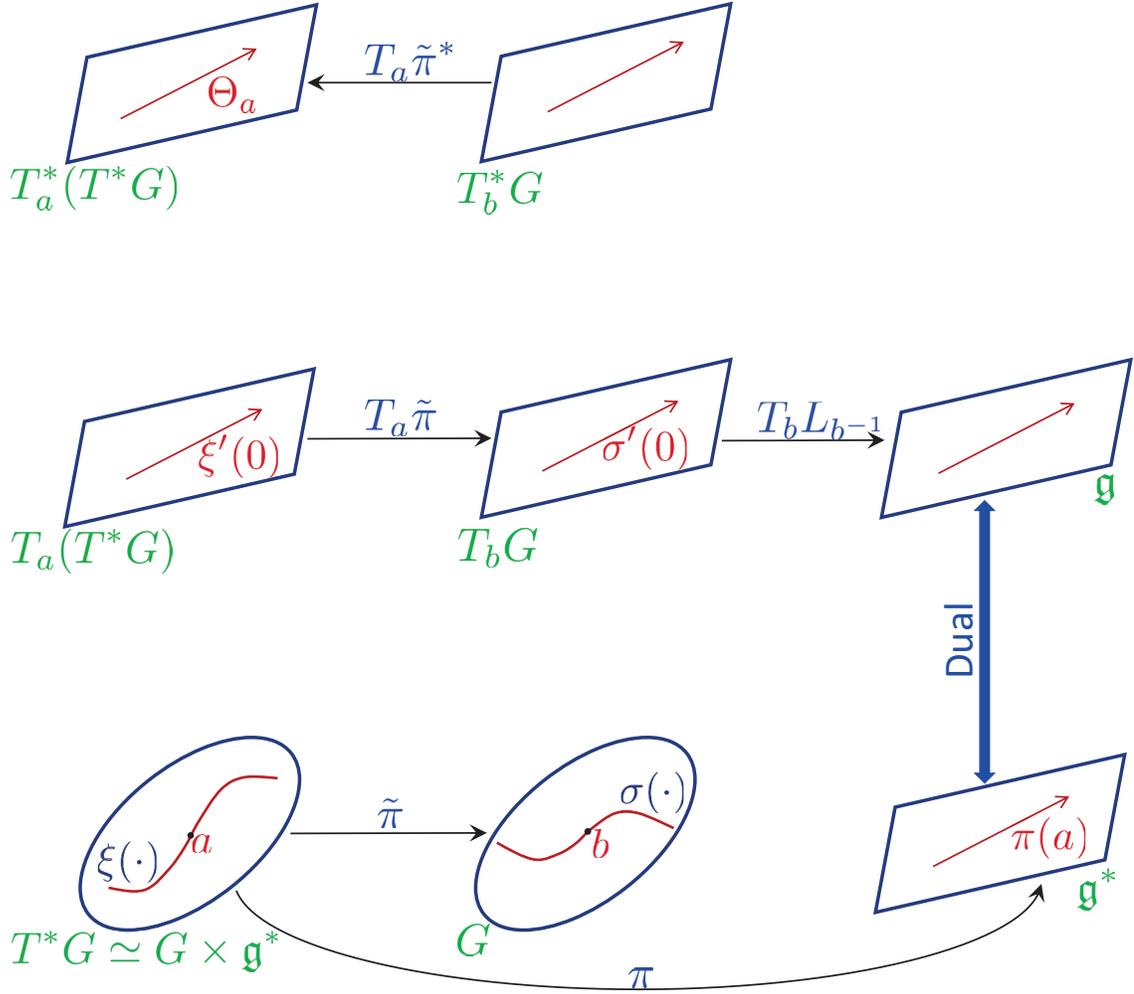


Figure 11: This figure illustrates Lie-Poisson Reduction.

where π^* denotes the pullback by π and $h_1, h_2 \in \mathcal{C}^\infty(\mathfrak{g}^*)$.

Next, we introduce $\mu = T_e L_{g^*}^* \cdot p$ to represent the *dual control curve* on \mathfrak{g}^* and formalize the mapping of the integral curve (g^*, p) of a left invariant Hamiltonian vector field onto the dual of the Lie algebra. By letting $\{X_i^b\}_{i=1}^n$ denote the dual basis for \mathfrak{g}^* , μ can be represented as $\mu = \sum_{i=1}^n \mu_i X_i^b$. On the other hand, a left-invariant Hamiltonian $H(g^*, p, u^*)$ projects to a reduced Hamiltonian (h) on \mathfrak{g}^* . Now, h defines a Hamiltonian vector field through the Lie-Poisson bracket, thus defining the dynamics for μ . Finally, through an explicit computation of the Lie-Poisson

bracket, we can derive the reduced dynamics as

$$\dot{\mu}_i = - \sum_{j=1}^n \sum_{k=1}^n \mu_k \Gamma_{ij}^k \frac{\partial h}{\partial \mu_j}, \quad (4.61)$$

where Γ_{ij}^k denote the structure constants associated with the Lie algebra \mathfrak{g} .

4.4 Example I: Data Smoothing on $SO(3)$

First we consider a left-invariant dynamics on $SO(3)$ governed by

$$\begin{aligned} \dot{g} &= g(u_1 X_1 - u_2 X_2) = g \xi_u, \\ g &\in SO(3), \quad \xi_u \in \mathfrak{so}(3), \end{aligned} \quad (4.62)$$

where u_1, u_2 denote curvature control inputs (for curvature), and

$$X_1 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad X_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \quad X_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix},$$

define a basis for the associated Lie algebra $\mathfrak{so}(3)$. Clearly, by assuming only two controls for a system evolving on a three dimensional manifold, we have cast the problem in a sub-Riemannian setting.

Now we consider a data smoothing problem on $SO(3)$ which attempts to find a curve $g : [t_0, t_N] \rightarrow SO(3)$ to traverse (approximately) through the sequence of targeted orientations $g_0 \rightarrow g_1 \rightarrow \dots \rightarrow g_N$ at time t_0, t_1, t_2 , and so on, respectively.

Our approach imposes regularization to this inverse problem by trading total fit-error against high values of the curvature path integral, and therefore the smoothing problem can be expressed as the following optimal control problem:

$$\begin{aligned}
& \underset{g(t_0), u_1, u_2}{\text{Minimize}} && \sum_{i=0}^N \|\mathbb{I}_3 - g(t_i)g_i^T\|_F^2 + \lambda \int_{t_0}^{t_N} (u_1^2 + u_2^2) dt \\
& \text{subject to} && \dot{g} = g(u_1 X_1 - u_2 X_2), \\
& && g : [t_0, t_N] \rightarrow SO(3), \quad u_1, u_2 \in \mathcal{U},
\end{aligned} \tag{4.63}$$

where \mathcal{U} is the space of real valued functions on $[t_0, t_N]$ and \mathbb{I}_3 denotes a 3×3 identity matrix. $\lambda (> 0)$ is the regularization parameter which maintains a balance between goodness of fit and smoothness of the reconstructed trajectory on $SO(3)$. By comparing this optimal control problem (4.63) with the one mentioned in the statement of maximum principle (4.45) we have

$$\begin{aligned}
L(u) &= \lambda(u_1^2 + u_2^2) = \lambda \langle \xi_u, \xi_u \rangle_{\mathfrak{so}(3)} \\
F(g(t_i), g_i) &= \|\mathbb{I}_3 - g(t_i)g_i^T\|_F^2
\end{aligned}$$

where the inner-product on $\mathfrak{so}(3)$ is defined as $\langle v_1, v_2 \rangle_{\mathfrak{so}(3)} = \frac{1}{2} \text{Tr}(v_1^T v_2) = \frac{1}{2} \text{Tr}(v_1 v_2^T)$ for $v_1, v_2 \in \mathfrak{so}(3)$.

4.4.1 Maximum Principle

Restricting our attention to normal extremals, we define the pre-Hamiltonian as

$$H(g, p, u) = \langle p, T_e L_g \cdot \xi_u \rangle - L(u) \quad (4.64)$$

where $p \in T_g^* SO(3)$, and $T_e L_g$ represents the tangent lift of the left translation by a group element g on $SO(3)$. Now we introduce μ at the dual of the Lie algebra ($\mathfrak{so}^*(3)$), defined as $\mu = T_e L_g^* \cdot p$. By letting X_i^b , $i = 1, 2, 3$ denote a dual basis for $\mathfrak{so}^*(3)$ (corresponding to the primal basis $\{X_i\}_{i=1}^3$), μ can be represented as

$$\mu = \sum_{i=1}^3 \mu_i X_i^b.$$

Therefore, by exploiting left-invariance of the generative model (4.62), the pre-Hamiltonian can be expressed as

$$\begin{aligned} H(g, p, u) &= \langle T_e L_g^* \cdot p, \xi_u \rangle - L(u) = \left\langle \sum_{i=1}^3 \mu_i X_i^b, (u_1 X_1 - u_2 X_2) \right\rangle - L(u) \\ &= u_1 \mu_1 - u_2 \mu_2 - \lambda(u_1^2 + u_2^2). \end{aligned} \quad (4.65)$$

As both ξ_u and $L(u)$ are differentiable with respect to u , an optimal control input (u^*) can be obtained by solving

$$\left. \frac{\partial H}{\partial u_i} \right|_{u_i=u_i^*} = 0, \quad i = 1, 2. \quad (4.66)$$

Then (4.65) and (4.66) yield the optimal control inputs as

$$\begin{pmatrix} u_1^* \\ u_2^* \end{pmatrix} = \frac{1}{2\lambda} \begin{pmatrix} \mu_1 \\ -\mu_2 \end{pmatrix}, \quad (4.67)$$

and by substituting the optimal controls into the pre-hamiltonian, (4.65) yields an $SO(3)$ -invariant hamiltonian. Hence we have the reduced hamiltonian on $\mathfrak{so}^*(3)$, given by

$$h = \frac{1}{4\lambda}(\mu_1^2 + \mu_2^2). \quad (4.68)$$

4.4.2 Frechet Derivative of the Fit Error

On the other hand, we need to evaluate the Frechet derivative of the fit-error in order to compute the jump conditions for μ . By using the definition of Frobenius norm, the fit error (4.63) can be expressed as

$$\begin{aligned} F(g(t_i), g_i) &= \|\mathbb{I}_3 - g(t_i)g_i^T\|_F^2 \quad g(t_i), g_i \in SO(3) \\ &= \text{Tr} \left[(\mathbb{I}_3 - g(t_i)g_i^T)^T (\mathbb{I}_3 - g(t_i)g_i^T) \right] \\ &= 2 \text{Tr} [\mathbb{I}_3 - g_i g^T(t_i)]. \end{aligned} \quad (4.69)$$

Now we assume h to be a tangent vector at $g(t_i) \in SO(3)$, and hence h can be parametrized as $h = g(t_i)\phi$ where $\phi^T = -\phi \in \mathfrak{so}(3)$. Therefore we have

$$\begin{aligned} F(g(t_i)e^{\epsilon\phi}, g_i) &= 2 \operatorname{Tr} [\mathbb{I}_3 - g_i(e^{\epsilon\phi})^T g^T(t_i)] \\ &= 2 \operatorname{Tr} [\mathbb{I}_3 - g_i(\mathbb{I}_3 + \epsilon\phi^T + O(\epsilon^2))g^T(t_i)]. \end{aligned} \quad (4.70)$$

From (4.69) and (4.70) we can compute the Frechet differential of F along h in the following way

$$\begin{aligned} D_{g(t_i)}F(h) &= \lim_{\epsilon \rightarrow 0} \frac{1}{\epsilon} \left(F(g(t_i)e^{\epsilon\phi}, g_i) - F(g(t_i), g_i) \right) \\ &= \lim_{\epsilon \rightarrow 0} \frac{2}{\epsilon} \operatorname{Tr} \left(-\epsilon g_i \phi^T g^T(t_i) + O(\epsilon^2) \right) \\ &= -2 \operatorname{Tr} \left(g_i \phi^T g^T(t_i) \right) \\ &= 2 \operatorname{Tr} \left(g^T(t_i) g_i \phi \right). \end{aligned} \quad (4.71)$$

Now we recall the fact that for a skew-symmetric matrix $B = -B^T \in \mathbb{R}^{n \times n}$ we have

$$\operatorname{Tr}(AB) = \frac{1}{2} \operatorname{Tr}((A - A^T)B)$$

for any $A \in \mathbb{R}^{n \times n}$, and by using this fact the differential of $F : SO(3) \rightarrow \mathbb{R}$ (4.71)

can be represented as

$$\begin{aligned}
D_{g(t_i)}F(h) &= \text{Tr} \left([g^T(t_i)g_i - g_i^T g(t_i)]\phi \right) \\
&= \text{Tr} \left([g^T(t_i)g_i g^T(t_i) - g_i^T]h \right) \\
&= \langle 2(g(t_i)g_i^T g(t_i) - g_i), h \rangle_{g(t_i)},
\end{aligned} \tag{4.72}$$

and hence we have $D_{g(t_i)}F = 2(g(t_i)g_i^T g(t_i) - g_i)$. By noting that $h \in T_{g(t_i)}SO(3)$ can be expressed as $T_e L_{g(t_i)} \cdot \phi$, we get

$$T_e L_{g(t_i)}^* \cdot D_{g(t_i)}F = 2(g_i^T g(t_i) - g^T(t_i)g_i). \tag{4.73}$$

4.4.3 Reduced Dynamics and Jump Discontinuities on $\mathfrak{so}^*(3)$

Next we focus on the derivation of reduced dynamics and associated jump conditions. By following the path laid out by Krishnaprasad [1993], the reduced dynamics on $\mathfrak{so}^*(3)$ can be computed as

$$\dot{\mu}_i(t) = - \sum_{j=1}^3 \sum_{k=1}^3 \mu_k(t) \Gamma_{ij}^k \frac{\partial h}{\partial \mu_j}(t), \quad i = 1, 2, 3, \tag{4.74}$$

where the temporal variable t lies in the open intervals (t_l, t_{l+1}) , $l = 0, \dots, N-1$, and Γ_{ij}^k denote the structure constants associated with the Lie algebra $\mathfrak{so}(3)$. Moreover, the corresponding jump conditions for μ can be obtained via Frechet derivative of the fit-error.

By computing the corresponding Lie brackets on $\mathfrak{so}(3)$ as

$$[X_1, X_2] = -X_3, \quad [X_2, X_3] = -X_1, \quad [X_3, X_1] = -X_2, \quad (4.75)$$

the associated structure constants (the nonzero ones) can be expressed in the following way

$$\Gamma_{12}^3 = -1, \quad \Gamma_{23}^1 = -1, \quad \Gamma_{31}^2 = -1, \quad (4.76)$$

and $\Gamma_{ij}^k = -\Gamma_{ji}^k$, $1 \leq i, j, k \leq 3$. Then, by exploiting (4.96), the reduced dynamics on the dual of Lie algebra can be expressed as

$$\begin{pmatrix} \dot{\mu}_1 \\ \dot{\mu}_2 \\ \dot{\mu}_3 \end{pmatrix} = \frac{1}{2\lambda} \begin{pmatrix} \mu_2\mu_3 \\ -\mu_3\mu_1 \\ 0 \end{pmatrix}, \quad t \in (t_k, t_{k+1}), \quad (4.77)$$

along with the jump conditions

$$\begin{aligned} & \mu_i(t_k^+) - \mu_i(t_k^-) \\ &= \langle T_e L_{g(t_k)}^* \cdot D_{g(t_k)} F, X_i \rangle_{\mathbb{I}_3 \in SO(3)} \quad i = 1, 2, 3 \\ &= \langle 2(g_k^T g(t_k) - g^T(t_k)g_k), X_i \rangle_{\mathbb{I}_3 \in SO(3)} \\ &= \text{Tr} (g^T(t_k)g_k X_i - g_k^T g(t_k)X_i), \end{aligned} \quad (4.78)$$

where $k = 0, 1, \dots, N-1$.

4.4.4 Explicit Solution of the Reduced Dynamics

In what follows, we develop a closed form solution (involving trigonometric functions) for the reduced dynamics. Clearly, (4.77) yields an explicit solution of the form

$$\begin{aligned}\mu_1(t) &= A_k \sin(C_k t + \phi_k) \\ \mu_2(t) &= A_k \cos(C_k t + \phi_k) \quad t \in (t_k, t_{k+1}), \\ \mu_3(t) &= 2\lambda C_k\end{aligned}\tag{4.79}$$

where the piecewise constant parameters A_k , C_k and ϕ_k can be computed using the boundary values and jump conditions of μ . As optimality ($p(t_0^-) = 0$) causes $\mu(t_0^-)$ to be equal to 0, we can compute the initial values of the solution parameters (C_k, A_k, ϕ_k) as

$$\begin{aligned}C_0 &= \frac{1}{2\lambda} \operatorname{Tr} \left((g^T(t_0)g_0 - g_0^T g(t_0)) X_3 \right) \\ A_0 &= 2\sqrt{\left(\operatorname{Tr} (\mathbb{I}_3 - (g^T(t_0)g_0)^2) \right)^2 - \lambda^2 C_0^2} \\ \phi_0 &= \operatorname{atan2} \left(\operatorname{Tr} ((g^T(t_0)g_0 - g_0^T g(t_0)) X_2), \operatorname{Tr} ((g^T(t_0)g_0 - g_0^T g(t_0)) X_1) \right) - C_0 t_0.\end{aligned}\tag{4.80}$$

Now we focus on the jump conditions for the solution parameters. Clearly, the jumps in μ_3 can be translated to an equivalent condition for C_k , given by

$$C_k - C_{k-1} = \frac{1}{2\lambda} \operatorname{Tr} \left((g^T(t_k)g_k - g_k^T g(t_k)) X_3 \right),\tag{4.81}$$

where $k \in \{1, 2, \dots, N-1\}$. However, there is no straightforward way to represent the jump discontinuities in A_k and ϕ_k . Instead, the following equations should be used to update these parameters

$$\begin{aligned} A_k \sin(C_k t_k + \phi_k) &= \text{Tr} \left((g^T(t_k)g_k - g_k^T g(t_k)) X_1 \right) + A_{k-1} \sin(C_{k-1} t_k + \phi_{k-1}) \\ A_k \cos(C_k t_k + \phi_k) &= \text{Tr} \left((g^T(t_k)g_k - g_k^T g(t_k)) X_2 \right) + A_{k-1} \cos(C_{k-1} t_k + \phi_{k-1}). \end{aligned}$$

Finally, the terminal value of the costate variable yields the following terminal condition for the solution parameters:

$$\begin{aligned} C_{N-1} &= -\frac{1}{2\lambda} \text{Tr} \left((g^T(t_N)g_N - g_N^T g(t_N)) X_3 \right) \\ A_{N-1} &= 2\sqrt{\left(\text{Tr} \left(\mathbb{I}_3 - (g^T(t_N)g_N)^2 \right) \right)^2 - \lambda^2 C_{N-1}^2} \\ \phi_{N-1} &= \text{atan2} \left(\text{Tr} \left((g^T(t_N)g_N - g_N^T g(t_N)) X_2 \right), \text{Tr} \left((g^T(t_N)g_N - g_N^T g(t_N)) X_1 \right) \right) \\ &\quad - C_{N-1} t_N. \end{aligned} \tag{4.82}$$

It is clear at this point that the optimal control inputs (4.67), along with their boundary values and intermediate jump conditions, can be evaluated using (4.79)-(4.82).

As the sinusoidal optimal control inputs are in phase quadrature, they can be interpreted as the natural curvatures for a circular helix, and hence it is possible to write down explicit solutions for the group dynamics on $SO(3)$ [Justh & Krishnaprasad, 2011]. Finally, an optimal initial condition is selected in such way that the terminal values of the solution parameters (4.82) are consistent with their

initial values (4.80) and intermediate update rules (4.82). Thus, we have turned an optimal control problem over an infinite dimensional space ($SO(3) \times \mathcal{U} \times \mathcal{U}$) into a two-point boundary value problem, which can be tackled by adopting an appropriate multiple-shooting method [Morrison *et al.*, 1962].

4.5 Example II: Data Smoothing on $SE(2)$

We begin our discussion about data smoothing on $SE(2)$ by considering the dynamics of a unicycle moving on a plane. By letting $(x, y) \in \mathbb{R}^2$ and $\theta \in \mathcal{S}^1$ denote position and heading angle of the unicycle, the underlying dynamics can be expressed as

$$\begin{aligned}\dot{x}(t) &= u_1(t) \cos \theta(t) \\ \dot{y}(t) &= u_1(t) \sin \theta(t) \\ \dot{\theta}(t) &= u_2(t)\end{aligned}\tag{4.83}$$

where u_1 and u_2 denote the speed and steering rate, respectively. These equations pose a nonholonomic constraint on the system, namely $\dot{x} \sin \theta = \dot{y} \cos \theta$, prohibiting any side-slip of the unicycle. Alternatively, by packing the position vector (x, y) , along with the heading direction (disguised through $\cos \theta, \sin \theta$), inside a 3×3 matrix

$g(t)$ defined as

$$g(t) = \begin{bmatrix} \cos \theta(t) & -\sin \theta(t) & x(t) \\ \sin \theta(t) & \cos \theta(t) & y(t) \\ 0 & 0 & 1 \end{bmatrix},$$

the dynamics of the unicycle (4.83) can be expressed as a left-invariant dynamics on $SE(2)$. Now, by letting

$$X_1 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad X_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad X_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix},$$

denote a basis for the associated Lie algebra $\mathfrak{se}(2)$, the dynamics of the unicycle

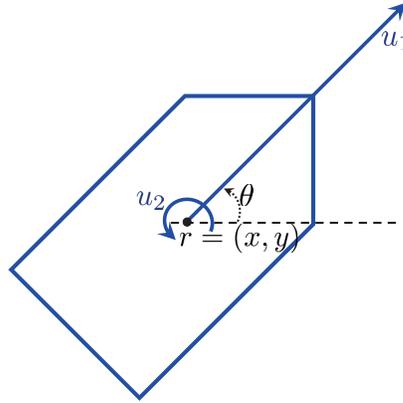


Figure 12: This figure illustrates the state variables associated to the dynamics of an unicycle.

can be represented as

$$\dot{g} = g(u_2 X_1 + u_1 X_2) = g \xi_u, \quad g \in SE(2), \quad (4.84)$$

where ξ_u defines a control curve on the Lie algebra.

Given a set of planar positions $\{r_i\}_{i=0}^N \subset \mathbb{R}^2$, we focus on finding a curve $g : [t_0, t_N] \rightarrow SE(2)$ which would traverse (approximately) through the sequence of targeted positions $r_0 \rightarrow r_1 \rightarrow \dots \rightarrow r_N$ at time t_0, t_1, t_2 , and so on, respectively. Our approach imposes regularization to this inverse problem by trading total fit-error against high values of the sum of speed and steering path integrals, and therefore the data smoothing problem can be expressed as the following optimal control problem on $SE(2)$:

$$\begin{aligned} & \underset{g(t_0), u_1, u_2}{\text{Minimize}} && \sum_{i=0}^N \|r(t_i) - r_i\|^2 + \lambda \int_{t_0}^{t_N} (u_1^2 + u_2^2) dt \\ & \text{subject to} && \dot{g} = g(u_2 X_1 + u_1 X_2), \\ & && g : [t_0, t_N] \rightarrow SE(2), \quad u_1, u_2 \in \mathcal{U}, \end{aligned} \tag{4.85}$$

where \mathcal{U} is the space of real valued functions on $[t_0, t_N]$, and $\lambda (> 0)$ maintains the balance between goodness of fit and smoothness of the path. This data smoothing problem can also be interpreted as a trajectory reconstruction problem for planar curves, where regularization is imposed by penalizing high values of the speed and steering rate path integrals.

Now, by comparing this data smoothing problem (4.85) with the optimal control problem mentioned in maximum principle statement (4.45), we have

$$\begin{aligned} L(u) &= \lambda(u_1^2 + u_2^2) = \lambda \langle \xi_u, \xi_u \rangle_{\mathfrak{se}(2)} \\ F(g(t_i), r_i) &= \|Ag(t_i)e_3 - r_i\|^2 \end{aligned}$$

where $A = [e_1 \ e_2]^T$, and $\{e_i\}_{i=1}^3$ denotes a standard basis vector in \mathbb{R}^3 . Moreover,

the inner-product on $\mathfrak{se}(2)$ is defined as $\langle v_1, v_2 \rangle_{\mathfrak{se}(2)} = \text{Tr}(v_1 M v_2^T)$ for $v_1, v_2 \in \mathfrak{se}(2)$,

where

$$M = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

is a symmetric, positive definite matrix.

4.5.1 Maximum Principle

Restricting our attention to normal extremals, we define the pre-Hamiltonian as

$$H(g, p, u) = \langle p, T_e L_g \cdot \xi_u \rangle - L(u) \quad (4.86)$$

where $p \in T_g^* SE(2)$, and $T_e L_g$ represents tangent lift of the left translation by a group element g on $SE(2)$. Now we introduce μ at the dual of the Lie algebra ($\mathfrak{se}^*(2)$), defined as $\mu = T_e L_g^* \cdot p$. By letting X_i^b , $i = 1, 2, 3$ denote a dual basis for $\mathfrak{se}^*(2)$ (corresponding to the primal basis $\{X_i\}_{i=1}^3$), μ can be represented as

$$\mu = \sum_{i=1}^3 \mu_i X_i^b.$$

Therefore, by exploiting left-invariance of the underlying dynamics (4.84), the pre-hamiltonian can be expressed as

$$\begin{aligned} H(g, p, u) &= \langle T_e L_g^* \cdot p, \xi_u \rangle - L(u) = \left\langle \sum_{i=1}^3 \mu_i X_i^b, (u_2 X_1 + u_1 X_2) \right\rangle - L(u) \\ &= u_2 \mu_1 + u_1 \mu_2 - \lambda(u_1^2 + u_2^2). \end{aligned} \quad (4.87)$$

As both ξ_u and $L(u)$ are differentiable with respect to u , an optimal control input (u^*) can be obtained by solving

$$\left. \frac{\partial H}{\partial u_i} \right|_{u_i=u_i^*} = 0, \quad i = 1, 2. \quad (4.88)$$

Then, from (4.87) and (4.88), the optimal control inputs can be expressed as

$$\begin{pmatrix} u_1^* \\ u_2^* \end{pmatrix} = \frac{1}{2\lambda} \begin{pmatrix} \mu_2 \\ \mu_1 \end{pmatrix}, \quad (4.89)$$

and by substituting the optimal controls into the pre-hamiltonian (4.87) yields an $SE(2)$ -invariant hamiltonian. Hence we have the reduced hamiltonian on $\mathfrak{se}^*(2)$, given by

$$h = \frac{1}{4\lambda}(\mu_1^2 + \mu_2^2). \quad (4.90)$$

4.5.2 Frechet Derivative of the Fit Error

Now we shift our attention towards computing the Frechet derivative of the fit-error. It is easy to check, that the fit error between data and the reconstructed

position (4.85) can be expressed as

$$\begin{aligned}
F(g(t_i), r_i) &= \|Ag(t_i)e_3 - r_i\|^2 \quad g(t_i) \in SE(2), \quad r_i \in \mathbb{R}^2 \\
&= e_3^T g^T(t_i) A^T Ag(t_i) e_3 - 2r_i^T Ag(t_i) e_3 + r_i^T r_i.
\end{aligned} \tag{4.91}$$

Now we assume h to be a tangent vector at $g(t_i) \in SE(2)$, and hence it can be parametrized as $h = g(t_i)\phi$ where $\phi \in \mathfrak{se}(2)$. Therefore, a perturbed value of the fit-error (at time t_i) can be expressed as

$$\begin{aligned}
F(g(t_i)e^{\epsilon\phi}, g_i) &= e_3^T (e^{\epsilon\phi})^T g^T(t_i) A^T Ag(t_i) (e^{\epsilon\phi}) e_3 - 2r_i^T Ag(t_i) (e^{\epsilon\phi}) e_3 + r_i^T r_i \\
&= e_3^T (\mathbb{I}_3 + \epsilon\phi^T + O(\epsilon^2)) g^T(t_i) A^T Ag(t_i) (\mathbb{I}_3 + \epsilon\phi + O(\epsilon^2)) e_3 \\
&\quad - 2r_i^T Ag(t_i) (\mathbb{I}_3 + \epsilon\phi + O(\epsilon^2)) e_3 + r_i^T r_i.
\end{aligned} \tag{4.92}$$

From (4.91) and (4.92) we can compute the Frechet differential of F along h as

$$\begin{aligned}
D_{g(t_i)} F(h) &= \lim_{\epsilon \rightarrow 0} \frac{1}{\epsilon} \left(F(g(t_i)e^{\epsilon\phi}, g_i) - F(g(t_i), g_i) \right) \\
&= \lim_{\epsilon \rightarrow 0} \frac{1}{\epsilon} \left[2\epsilon e_3^T \phi^T g^T(t_i) A^T Ag(t_i) e_3 - 2\epsilon r_i^T Ag(t_i) \phi e_3 + O(\epsilon^2) \right] \\
&= 2[Ag(t_i)e_3 - r_i]^T [Ahe_3] \\
&= 2 \operatorname{Tr} \left([Ag(t_i)e_3 - r_i][Ahe_3]^T \right) \\
&= 2 \operatorname{Tr} \left(A^T [Ag(t_i)e_3 - r_i] e_3^T h^T \right) \\
&= \langle 2A^T [Ag(t_i)e_3 - r_i] e_3^T M^{-1}, h \rangle_{g(t_i)},
\end{aligned} \tag{4.93}$$

and hence we have $D_{g(t_i)}F = 2A^T[Ag(t_i)e_3 - r_i]e_3^T M^{-1}$. Alternatively, by noting that the differential can be expressed as

$$\begin{aligned}
D_{g(t_i)}F(h) &= 2 \operatorname{Tr} \left(A^T [Ag(t_i)e_3 - r_i] e_3^T \phi^T g(t_i)^T \right) \\
&= 2 \operatorname{Tr} \left(g(t_i)^T A^T [Ag(t_i)e_3 - r_i] e_3^T \phi^T \right) \\
&= \langle 2g(t_i)^T A^T [Ag(t_i)e_3 - r_i] e_3^T M^{-1}, h \rangle_{g(t_i)}, \tag{4.94}
\end{aligned}$$

it can be concluded that

$$T_e L_{g(t_i)}^* \cdot D_{g(t_i)}F = 2g(t_i)^T A^T [Ag(t_i)e_3 - r_i] e_3^T M^{-1}. \tag{4.95}$$

4.5.3 Reduced Dynamics and Jump Discontinuities on $\mathfrak{se}^*(2)$

Next we focus on the derivation of reduced dynamics and associated jump conditions. By following the path laid out by Krishnaprasad [1993], the reduced dynamics on $\mathfrak{se}^*(2)$ can be computed as

$$\dot{\mu}_i(t) = - \sum_{j=1}^3 \sum_{k=1}^3 \mu_k(t) \Gamma_{ij}^k \frac{\partial h}{\partial \mu_j}(t), \quad i = 1, 2, 3, \tag{4.96}$$

where the temporal variable t lies in the open intervals (t_l, t_{l+1}) , $l = 0, \dots, N-1$, and Γ_{ij}^k denote the structure constants associated with the Lie algebra $\mathfrak{se}(2)$. Moreover, the corresponding jump conditions for μ can be obtained via Frechet derivative of the fit-error.

By computing the Lie brackets on $\mathfrak{se}(2)$ as

$$[X_1, X_2] = X_3, \quad [X_2, X_3] = 0, \quad [X_3, X_1] = X_2, \quad (4.97)$$

the associated structure constants can be expressed in the following way

$$\Gamma_{12}^3 = 1, \quad \Gamma_{31}^2 = 1, \quad (4.98)$$

with $\Gamma_{ij}^k = -\Gamma_{ji}^k$, $1 \leq i, j, k \leq 3$, and the rest of the structure constants are zero. As a result, the reduced dynamics on $\mathfrak{se}^*(2)$ can be expressed as

$$\begin{pmatrix} \dot{\mu}_1 \\ \dot{\mu}_2 \\ \dot{\mu}_3 \end{pmatrix} = \frac{1}{2\lambda} \begin{pmatrix} -\mu_2\mu_3 \\ \mu_3\mu_1 \\ -\mu_1\mu_2 \end{pmatrix} \quad t \in (t_k, t_{k+1}), \quad (4.99)$$

and the corresponding jump conditions are given by

$$\begin{aligned} & \mu_i(t_k^+) - \mu_i(t_k^-) \\ &= \langle T_e L_{g(t_k)}^* \cdot D_{g(t_k)} F, X_i \rangle_{\mathbb{I}_3 \in SE(2)} \quad i = 1, 2, 3 \\ &= \langle 2g(t_k)^T A^T [Ag(t_k)e_3 - r_k] e_3^T M^{-1}, X_i \rangle_{\mathbb{I}_3 \in SE(2)} \\ &= \text{Tr} (2g(t_k)^T A^T [Ag(t_k)e_3 - r_k] e_3^T X_i^T), \end{aligned} \quad (4.100)$$

where $k = 0, 1, \dots, N - 1$.

4.5.4 Explicit Solution of the Reduced Dynamics

Now we attempt to obtain a closed form solution for the reduced dynamics (4.99). It is easy to check that the reduced Hamiltonian (4.90) is a conserved quantity. Furthermore, by introducing

$$C = \frac{1}{4\lambda}(\mu_2^2 + \mu_3^2), \quad (4.101)$$

we can show that C is also conserved along the trajectories of (4.99). Then, by exploiting the constants of motion, namely the reduced Hamiltonian h (4.99) and the Casimir C (4.101), the dynamics of μ_2 can be represented as

$$\begin{aligned} \dot{\mu}_2 &= \frac{1}{2\lambda} \sqrt{(4\lambda h - \mu_2^2)(4\lambda C - \mu_2^2)} \\ &= 2\sqrt{hC} \sqrt{\left(1 - \frac{\mu_2^2}{4\lambda h}\right) \left(1 - \frac{h}{C} \frac{\mu_2^2}{4\lambda h}\right)}. \end{aligned} \quad (4.102)$$

Then it is straightforward to show that (4.102) yields an explicit solution involving Jacobi's elliptic sine function. Whenever $h \leq C$, the solution can be expressed as

$$\mu_2(t) = 2\sqrt{\lambda h} \operatorname{Sn} \left(\sqrt{\frac{C}{\lambda}}(t + \phi_k), \sqrt{\frac{h}{C}} \right), \quad (4.103)$$

where $Sn(\cdot, \cdot)$ denotes Jacobi's elliptic sine function and $t \in (t_k, t_{k+1})$. Now, by exploiting the standard identities for elliptic functions, we can express μ_1 and μ_3 as

$$\begin{aligned}\mu_1^2 &= 4\lambda h \operatorname{Cn}^2 \left(\sqrt{\frac{C}{\lambda}}(t + \phi_k), \sqrt{\frac{h}{C}} \right) \\ \mu_3^2 &= 4\lambda C \operatorname{Dn}^2 \left(\sqrt{\frac{C}{\lambda}}(t + \phi_k), \sqrt{\frac{h}{C}} \right),\end{aligned}\tag{4.104}$$

and the appropriate signs will depend on the initial/boundary conditions. A similar solution exists for the situation when $h > C$. Now, by exploiting the fact that both $\mu(t_0^-)$ and $\mu(t_N^+)$ are equal to zero, we turn it into a two-point boundary value problem, and solve it via an appropriate multiple-shooting method [Morrison *et al.*, 1962]. Finally, an optimal state trajectory can be obtained by integrating the group dynamics along with an optimal initial condition and optimal control inputs.

4.6 Conclusion

In this chapter, we have developed a framework (based on a modified version of the maximum principle) to solve data smoothing in a semi-analytic way, and our results are applicable to problems in both Euclidean and finite dimensional matrix Lie group settings. We demonstrate the pertinence of this approach by solving an example problem, wherein the generative model is governed by a left invariant vector field on a matrix Lie group ($SE(2)$) and regularization is imposed through a left invariant path cost (Lagrangian). In this special case Lie-Poisson reduction leading to explicitly integrable dynamics brings in further simplification into the problem, and we get closed-form solutions (in terms of Jacobi's elliptic functions).

Part II

Analysis of Collectives

Chapter 5: Analysis of Bat Foraging in Two Different Contexts

Research in bat echolocation has yielded a rich trove of insight into how the bat perceives the world through active acoustic probing (Griffin *et al.* [1960]; Simmons *et al.* [1979]). Yet surprisingly little is known about how perception is turned into action such as steering towards a target in foraging. Limited availability of suitable high speed motion capture technology hindered earlier efforts to address this question. More recently, trajectory analysis in the work by Ghose *et al.* [2006] showed that the big brown bat, *Eptesicus fuscus*, essentially maintains constant absolute target direction (CATD) while chasing a free flying target, a species of praying mantis (*Parasphendale agrionina*). However, in a later study of competitive foraging for a single tethered food source [Chiu *et al.*, 2010], evidence emerged that a big brown bat resorts to directing its flight towards a competitor, thus employing classical pursuit (CP).

In this present work¹, using geometric and statistical analysis and control theory, we describe a comparative study of how foraging context shapes bat flight strategy. The current study examines two different flight control strategies, namely classical pursuit (CP) and constant absolute target direction (CATD). The CP strat-

¹A significant portion of this chapter has been reproduced verbatim from a pre-print by Dey *et al.* [2014].

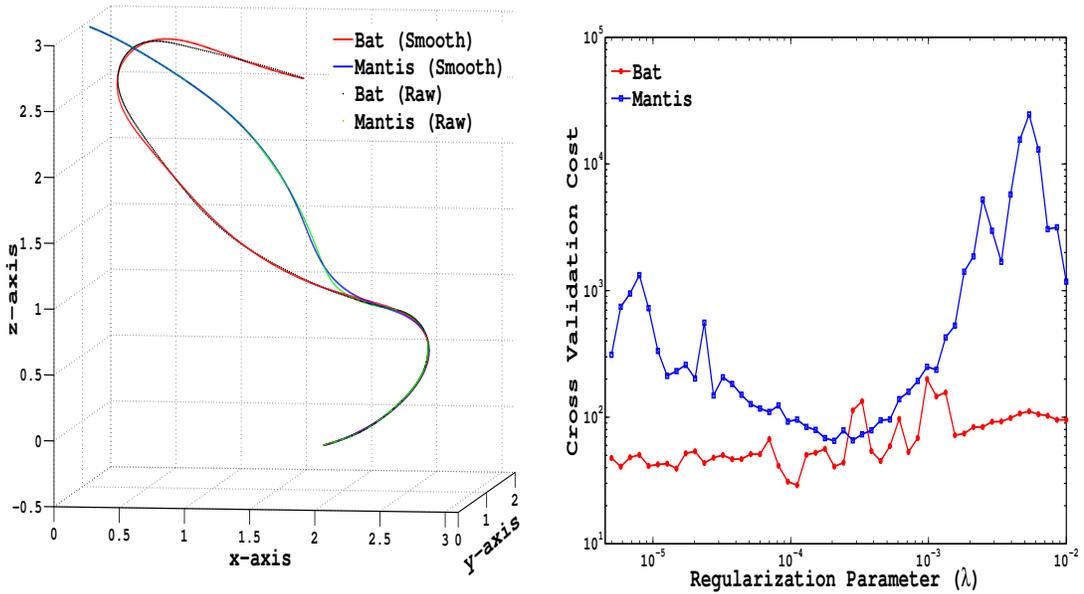
egy refers to a configuration in which the follower always points its velocity vector towards the position of a target (Galloway *et al.* [2010]; Wei *et al.* [2009]). The CP strategy can also be viewed as a special case of constant bearing (CB) pursuit strategy, which has gaze heuristic [McBeath *et al.*, 1995] as one of its manifestation. On the other hand, CATD is a stealth strategy in which the follower approaches the target in such a manner that from the target's point of view the follower always appears to be at the same bearing [Srinivasan & Davey, 1995]. This strategy is also known as motion camouflage (MC) strategy as it nullifies the transverse component of the relative velocity, and therefore the follower's optic flow vanishes to zero in the target's field of vision [Justh & Krishnaprasad, 2006]. This work also provides some insight about the behavioral latency associated with a bat-flight.

In one of the earliest works by Ghose & Moss [2006], investigating sensorimotor transformation in a foraging bat, it was possible to simultaneously record what the bat perceived through its acoustic gaze, and its response through steering action. The behavioral context here - a single bat trained to seek, localize and capture a tethered mealworm hanging from the ceiling of a darkened flight room - is simpler than that in either of the two later studies (Chiu *et al.* [2010]; Ghose *et al.* [2006]) that constitute the focus of the present paper. The resulting bat flight trajectories are essentially planar. A main result of the study by Ghose & Moss [2006] is the discovery of a feedback law that relates planar turning rate of the bat to acoustic gaze angle, and the observation that the gain parameter in the law is modulated by the echolocation pulse production rate (PPR) - higher the PPR, greater is the gain. This observed adaptive linkage of flight motor output to spatial auditory

information about the target (stationary food item) was suggested as a compromise between uncoupling gaze direction and flight control (possibly conserving energy) and tight coupling to ensure accuracy needed for successful taking of the food item. While not emphasized by Ghose & Moss [2006], the feedback law is consistent with executing what we call the CP strategy in this paper, thus indicating that the foraging bat has CP strategy in its repertoire of flight behavior. Besides CP and CATD other strategies such as following a boundary may also be identifiable as being part of the repertoire of an echolocating bat.

5.1 Experiment Details and Reconstruction of Trajectories

In this work we have analyzed the flight data collected from a series of experiments conducted by students of Prof. Cynthia Moss at the Auditory Neuroethology Laboratory (BATLAB), Department of Psychology, University of Maryland, and performed a comparative analysis of the underlying flight strategies and feedback mechanism for steering control. The bat flight experiments were carried out in a large flight room with two high-speed cameras (Kodak MotionCorder CCD-based cameras, running at 240 frames/s) placed in adjacent corners of the room. A commercially available motion analysis software (Motus, Peak Performance Technologies, Englewood, CO) was used to recover 3-D position data from high-speed stereo images. Finally we use the regularized inversion approach (Section 2.1) to smoothen the individual trajectories and extract their speed and curvatures. Furthermore, all animal care and experimental procedures were approved by the Institutional Animal



(a) A typical example of trajectory reconstruction.

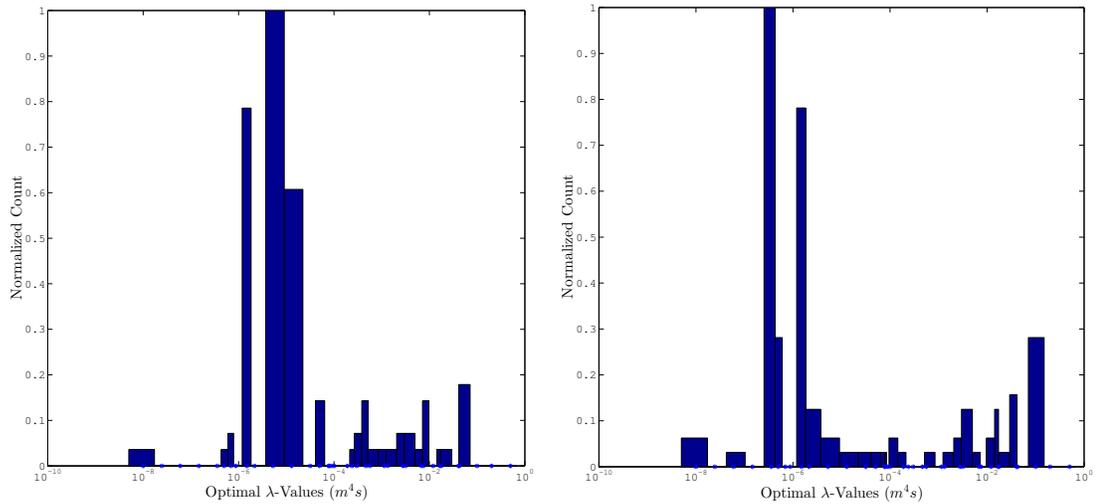
(b) The corresponding variation in OCV cost.

Figure 13: This figure shows reconstructed trajectories for a particular pursuit event along the with raw data. It also shows the variation of cross validation cost as a function of the smoothing parameter.

Care and Use Committee at the University of Maryland, College Park.

At the initial stage of our study of bat-mantis interactions, eight bats were trained to catch both free flying and tethered insects, and the room walls and ceiling of the indoor flight arena were covered with sound-absorbent foam. Later during the experiments, the praying mantis was released by hand as the bat was flying around in the room. Moreover the ultrasound triggered diving behavior of the mantises were suppressed by plugging Vaseline into their ears. For these bat-mantis experiments we analyzed 18 successful trials wherein the bat eventually catches the insect. As the number of trials were not very high we performed ordinary cross-validation (OCV) for each trajectory in each of the trials, and used the result of OCV in reconstructing the corresponding trajectory.

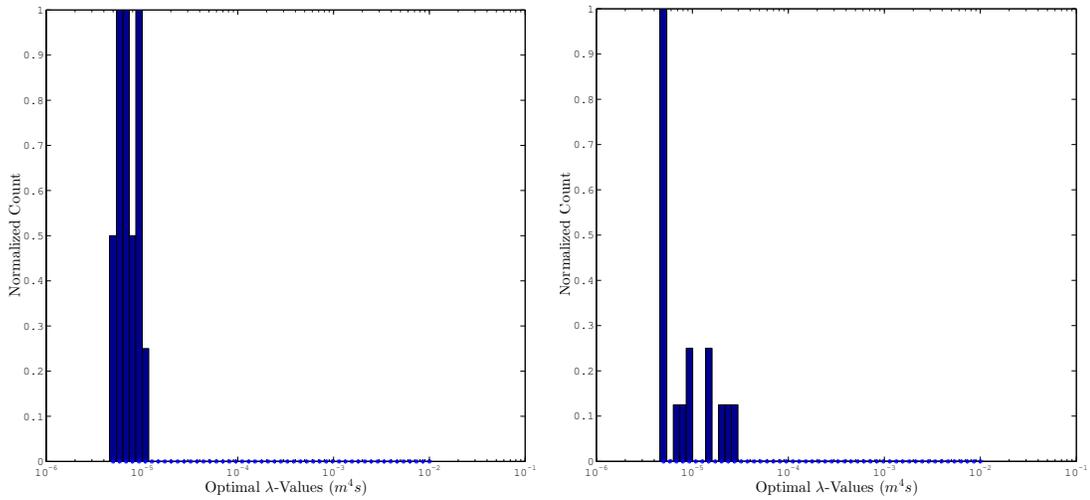
On the other hand, five big brown bats, forming four pairs, were used in the experiments for studying the bat-bat interactions. The experiments were conducted between July and September in 2005 and 2006. After being trained to capture a tethered mealworm individually, two bats were released simultaneously from the same spot in the flight room to compete for a single tethered mealworm. The trials ended whenever one bat made contact with the mealworm. For this set of bat-bat experiments we had 154 trials at our disposal. As the computational time for cross-validation and trajectory reconstruction is quite significant (around 2 days on a 12-core workstation, for a trajectory with 400 data points and 50 discrete values of λ -parameter), we decided to perform OCV and trajectory reconstruction over a subset of the whole data-set. This required extra care in picking the individual trials for reconstruction. So, we randomly selected 30 Bat-1 trajectories (through



(a) Distribution of optimal values of λ for Bat-2 trajectories. (b) Distribution of optimal values of λ for Bat-1 trajectories.

Figure 14: This figure shows the distribution of optimal values of regularization parameter (λ^*) used to reconstruct trajectories for bat-bat interactions.

a uniform random sampling) from a pool of 154 Bat-1 trajectories, and performed OCV on these trajectories. Then we chose the most frequent value of optimal λ for reconstructing Bat-1 trajectories. Similar approach was adopted to choose a λ -value for reconstructing Bat-2 trajectories. Next, we randomly selected 30 trials (out of 154) through a uniform random sampling, and this sampling was independent to the previous selections (for OCV). Finally we reconstructed the Bat-1 and Bat-2 trajectories for these 30 trials using the previously chosen λ -values.



(a) Distribution of optimal values of λ for bat trajectories. (b) Distribution of optimal values of λ for mantis trajectories.

Figure 15: This figure shows the distribution of optimal values of regularization parameter (λ^*) used to reconstruct trajectories for bat-mantis interactions.

5.2 Pre-processing of Trajectory Data

In the work by Ghose & Moss [2006], key aspect of data analysis is to recognize that flight behavior can be demarcated into segments. While the bat is initially ignorant of the location of the target (presented by opening a ceiling trapdoor at

a random location at random times), it goes through a sequence of stages from search/approach to tracking to attack of the mealworm, ending the trial. This demarcation of stages of flight was suggested by examination of histograms of PPR of the echolocation calls which show three clear peaks corresponding to the stages. Further in the course of a trial, increasing PPR is associated to increasing accuracy in localization of target prior to capture. The demarcation into stages was then used in the temporal segmentation of trajectory properties (turning rate, acoustic gaze angle etc.) for the purposes of statistical fitting of steering feedback laws in each segment.

In the setting of a single bat engaged in three-dimensional pursuit of a free-flying insect as in the work of Ghose *et al.* [2006], similar trajectory segmentation is possible, guided by echolocation PPR. In his dissertation, Reddy [2007] has used such segmentation in fitting feedback laws, focusing essentially on attack segments. In multiple trials highly maneuvering behavior is exhibited by the insect prey (whose sensitive hearing of bat sonar is (only partially) disabled to reduce the incidence of evasive action). Restricting data fitting mainly to segments characterized by PPR as attack segments misses important components of pursuit behavior taking place over the full course of a trial. In the setting of competitive prey capture the interaction between competing bats appears to be of great relevance to the outcome. During the course of such competitive interactions vocalization information is confounded by the presence of interspersed social calls to possibly communicate intent of one bat to another [Wright *et al.*, 2014]. There are also periods of silence (as observed by Chiu *et al.* [2008]) of one or more of the bats. Thus there is a need for disambiguation

between echolocation and social calls, appropriate for use in flight segmentation, whereas in the context of single bat prey capture flights this consideration does not arise. In this paper we side-step this disambiguation problem, by adopting an approach to segmentation that does not use vocalization patterns in either context, and instead is based on a geometric criterion described below.

In preparation for segmentation of trajectories as needed in the analysis of flight behavior, we introduce two geometric concepts, namely *following* and *convergence*, to assist us in understanding the individual roles in a dyadic (bat-mantis or bat-bat) interaction. By letting \mathbf{r}_i and \mathbf{x}_i (respectively \mathbf{r}_j and \mathbf{x}_j) denote the position and normalized velocity of the individual i (respectively the individual j) in a dyadic interaction, these two notions can be defined as:

Definition 5.1 (Following Property). *The flight behavior of an individual i (interacting with individual j) is called “following” when its velocity vector has a negative projection on its relative position vector \mathbf{r}_{ji} ($= \mathbf{r}_i - \mathbf{r}_j$), i.e.*

$$\frac{\mathbf{r}_{ji}}{|\mathbf{r}_{ji}|} \cdot \mathbf{x}_i < 0. \quad (5.1)$$

Definition 5.2 (Convergence). *The flight behavior of an individual i is called “converging towards the individual j ” if the distance between the individuals is shrinking, i.e.*

$$\frac{\mathbf{r}_{ji}}{|\mathbf{r}_{ji}|} \cdot \dot{\mathbf{r}}_{ji} < 0, \quad (5.2)$$

where the time derivative $\dot{\mathbf{r}}_{ji}$ denotes the relative velocity.

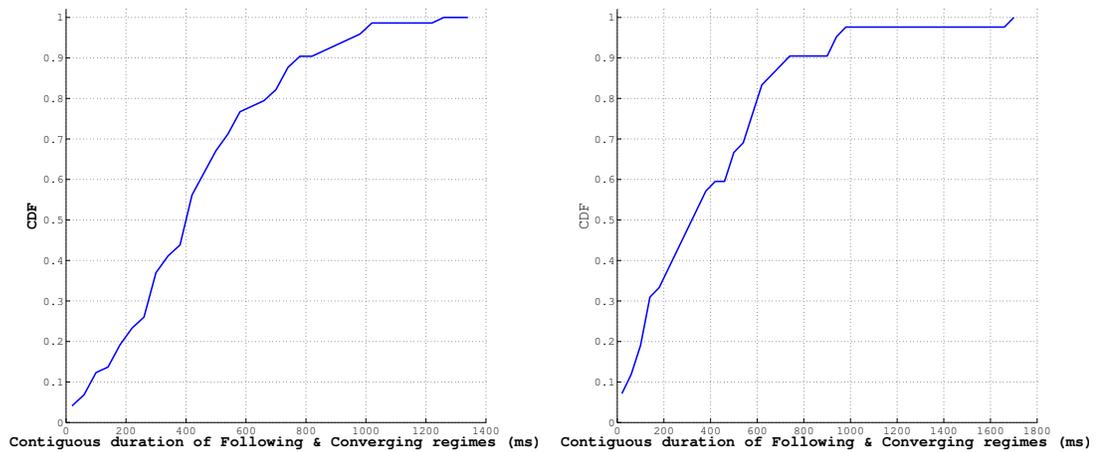
Clearly, these two notions split a flight into the following four different regimes:

- *Class I* : Following and Converging,
- *Class II* : Following but Not Converging,
- *Class III* : Not Following but Converging, and
- *Class IV* : Neither Following nor Converging,

depending on the angle between the baseline vector and the velocity vectors of the individuals. We should note that the *convergence* property is symmetric with respect to individual roles. Any particular flight trial will have a number of different contiguous segments characterized by the properties of following and convergence. In Fig 16, we display distribution functions across all trials of the durations of contiguous segments which are simultaneously following *and* converging. Averaging over multiple flight trials, we designate an individual as *follower (pursuer)* in a dyadic interaction, if it obeys the *following* condition for a duration longer than that for the other individual (which we designate as *pursuee*).

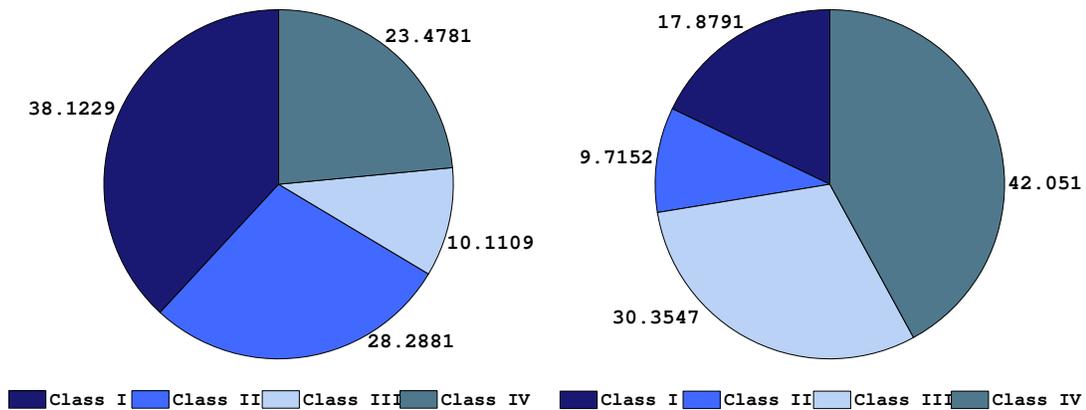
The pie-chart in Fig 17a shows the percentage of all the four classes in the flight data collected for Bat-2. From this pie-chart one can notice that Bat-2 follows Bat-1 for more than 66% of the time. On the contrary Bat-1 follows Bat-2 for only 27% of the time. Therefore we consider Bat-2 to be the follower of Bat-1 for further analysis. This methodology for classification reveals that in spite of the setup being symmetric there is a strong evidence of leader-follower relationship when a bat competes for a single food source with another conspecific.

On the other hand the pie-chart in Fig 18 shows the percentage of all four classes in the flight data from bat-mantis trajectory pairs. From this pie-chart it is



(a) Bat-bat interactions (Pursuer: Bat-2, Pursuee: Bat-1). (b) Bat-mantis interactions (Pursuer: Bat, Pursuee: Mantis).

Figure 16: This figure shows the CDF for contiguous durations of following and converging (Class I) flight segments.



(a) Percentage of four classes of flight behavior for the competitive interaction between two bats. (b) Effect of role reversal on the segmentation of flight behavior for the bat-bat interaction.

Figure 17: This figure illustrates the fact that in a competitive interaction between two big brown bats one of the bats leads the other.

clear that the bat follows the target for around 75% of the time. It also converges towards the praying mantis for more than 70% of the time.

5.3 Analysis of Flight Strategy

Once we have segmented each trajectory into four classes (*following and converging*; *following, but not converging*; *not following, but converging*; and *neither following nor converging*), we examine the trajectory data to weigh support for or against the pursuit strategies CP and CATD. This is done by computing contrast function values (Λ for CP and Γ for CATD/MC, Section 1.3.2) associated with the data, and comparing the value distributions. A value close to -1 for a contrast function means strong support for the associated strategy. On the other hand a value of $+1$ denotes maximum departure from the associated strategy. In order to determine how long a trailing bat remained in the CP or CATD state, we consider the duration of the flight when the contrast function value goes below -0.9 , i.e. it lies in the range $[-1, -0.9]$. This duration of interest can be easily computed as a percentage of the total flight duration by paying attention to the cumulative

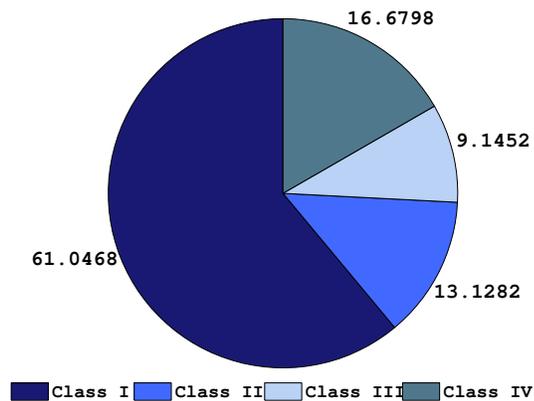


Figure 18: This figure shows the percentage of four flight behavior classes in bat-mantis pursuit events.

distribution of contrast function values. The rationale behind choosing the threshold at -0.9 is based on the principle that outcome of a statistical hypothesis test should not depend on a test parameter. It should be noted here that, as a means for closer inspection, we also restrict our focus on those flight regimes which are both *following and converging* (Class I), and recompute the statistics. This provides a common base for comparison, ensuring that neither of the contrast function values become non-negative.

Initially we analyze the bat-bat pursuit strategy for the complete set of reconstructed flight data, and the corresponding histograms of Λ and Γ are shown in Fig 19. The closer the contrast function is to -1 , the more the bats' flight behavior approaches a particular pursuit strategy. The peak of the CP contrast function is positioned around -1 , which indicates that the following bat mostly relies on the CP strategy to pursue the leader (another conspecific). On the other hand, the CATD contrast function is more evenly distributed between -1 and 1 , indicating lack of ev-

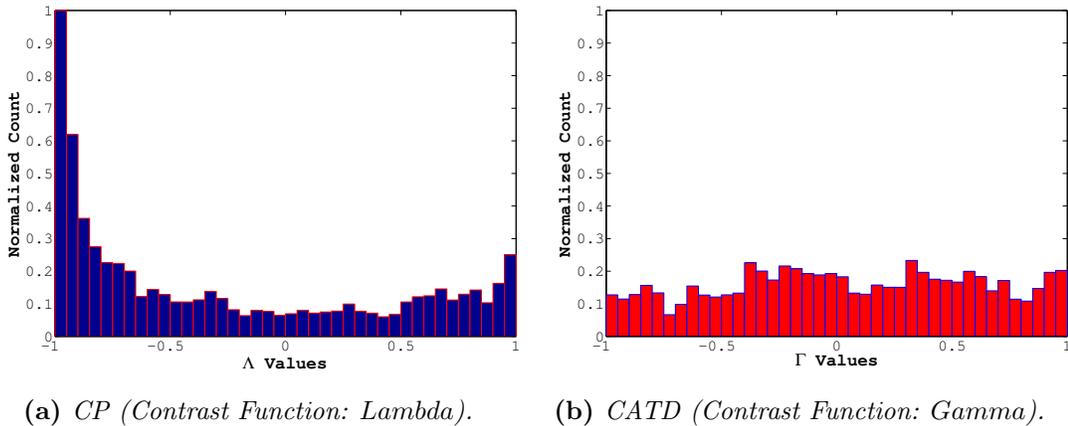
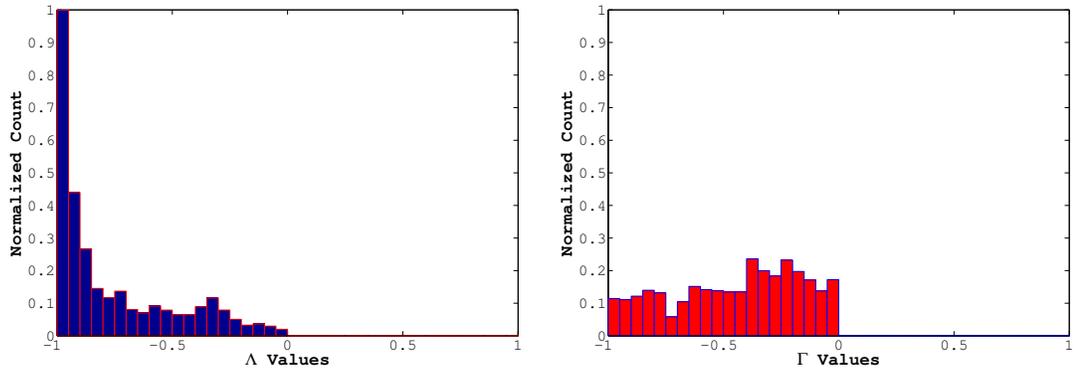


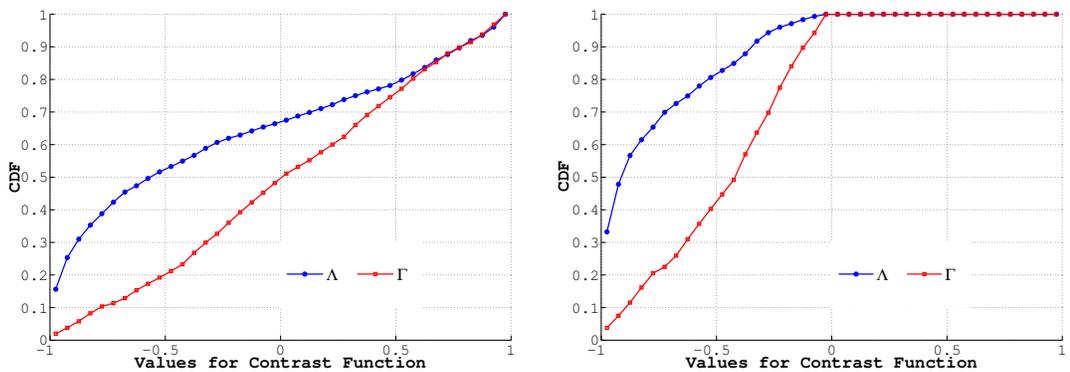
Figure 19: Distribution of contrast function values shows dominance of CP during bat-bat interactions. The frequency (y -axis) is normalized by the maximum count (considering both strategies).



(a) CP (Contrast Function: Lambda). (b) CATD (Contrast Function: Gamma).

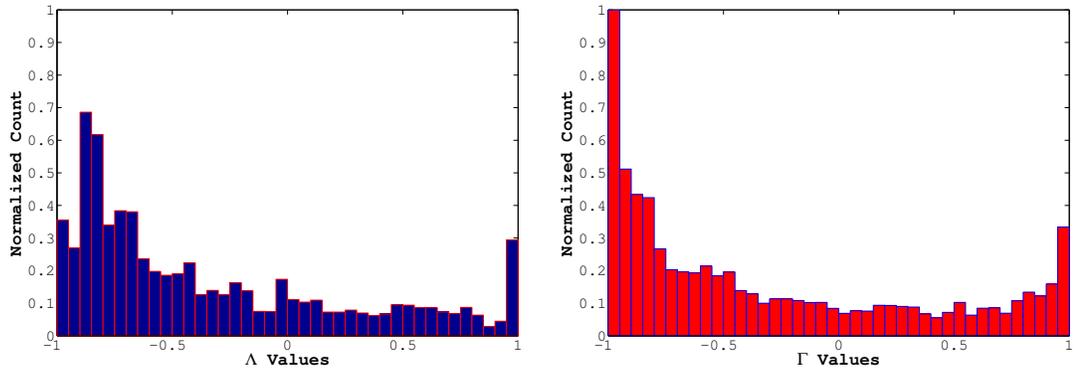
Figure 20: Distribution of Λ and Γ values shows dominance of CP during the following and converging segments (class I) of bat-bat pursuit events. The frequency (y-axis) is normalized by the maximum count (considering both strategies)

idence in favor of the CATD pursuit strategy. Moreover, the data show that 25.32% of the time the following bat stays in the CP state, while on contrary it stays in CATD state for only 3.79% of the time. This difference between evidence for each of the individual strategies becomes more prominent if we focus on the following and converging flight behavior, i.e. if data points from only class I are considered for



(a) Comparison of CDFs of the contrast functions for CP and CATD (whole data set). (b) Comparison of CDFs of the contrast functions for CP and CATD (during class I).

Figure 21: Comparison of CDFs for the contrast functions shows dominance of classical pursuit during bat-bat interactions.

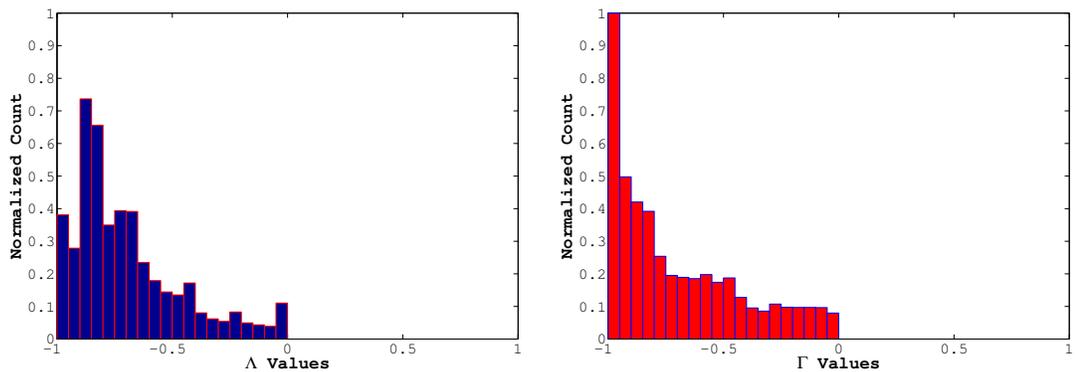


(a) *CP (Contrast Function: Lambda).* (b) *CATD (Contrast Function: Gamma).*

Figure 22: *Distribution of Λ and Γ shows dominance of CATD/MC during bat-mantis interactions. The frequency is normalized by the maximum count (considering both strategies).*

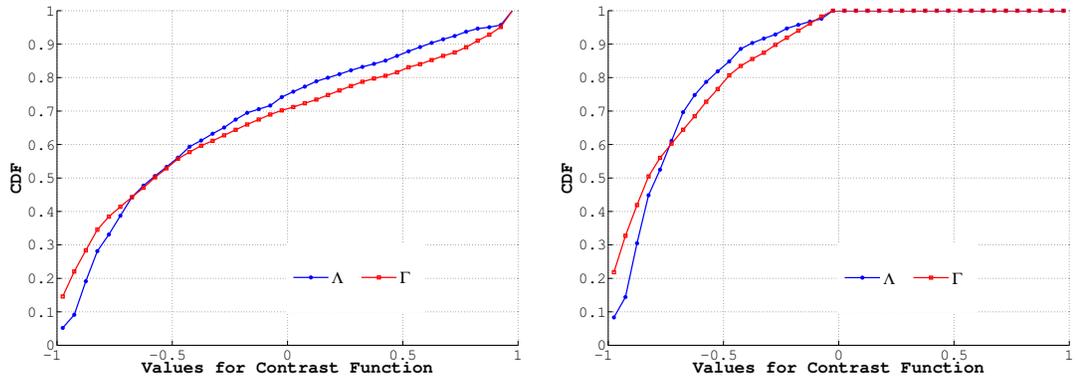
strategy analysis. In that case the following bat stays in the CP state for 47.83% of the time, against 7.48% of the time spent in the CATD state. The associated histograms are shown in Fig 20. Therefore our present work reconfirms the findings by Chiu *et al.* [2010], i.e. *bats do not apply CATD while pursuing conspecifics, rather they use CP strategy to follow another bat.*

On the other hand, Fig 22 shows the histograms of Λ and Γ for the complete



(a) *CP (Contrast Function: Lambda).* (b) *CATD (Contrast Function: Gamma).*

Figure 23: *Distribution of Λ and Γ shows dominance of CATD/MC during the following and converging segments (class I) of bat-mantis pursuit events. The frequency is normalized by the maximum count (considering both strategies)*



(a) Comparison of CDFs of the contrast functions for CP and CATD (whole data set). (b) Comparison of CDFs of the contrast functions for CP and CATD (during class I).

Figure 24: Comparison of CDFs for the contrast functions shows dominance of CATD/MC during bat-mantis interactions.

set of bat-mantis trajectory pairs. From this figure one can notice that the data indicates strong evidence in favor of CATD pursuit strategy. Quantitatively, the bat stays in the CATD state for 22.05% of the time in comparison to 9.12% of the time in the CP state. Now we narrow our focus to the following and converging segments of the flight, i.e. data points from only class I are considered for strategy analysis. In that case the bat stays in the CATD state for 32.72% of the time, against 14.42% of the time in the CP state (Fig 23). Therefore our analysis provides support for *bats' use of CATD for pursuing a free flying mantis*.

However, at this level of flight strategy analysis (through studying the distribution of associated contrast function values), the significance of distinction between CP and CATD is less sharply delineated for bat-mantis interactions than for bat-bat interactions. This leads us to the next part of our analysis, i.e. comparison at the level of feedback laws for steering control.

5.4 Analysis of Steering Control

A bat executes a pursuit strategy by continually adjusting the curvature of its trajectory based on its perception of the relative motion of the target. Such strategy-specific steering feedback control laws have been a subject of applied mathematical research (Galloway *et al.* [2010]; Justh & Krishnaprasad [2006]; Reddy [2007]; Reddy *et al.* [2006]). Here we investigate steering controls that underlie a bat’s pursuit strategy by comparing empirical values of trajectory curvature with predictions from theoretically well-founded feedback control laws (Table 5.1), and provide evidence at the level of the steering control mechanism as well.

By letting, (u_{em}, v_{em}) denote the empirical curvatures obtained from trajectory smoothing and (u_{th}, v_{th}) denote the curvature values computed using feedback laws indicated in Table 5.1, we formalize this analysis as the following *mismatch minimization* problem

$$\text{Minimize}_{\mu > 0, \delta \in \mathbb{N}} \left(\frac{1}{\sum_{j \in \mathcal{S}} (|\mathcal{E}_j| - \delta)} \sum_{j \in \mathcal{S}} \sum_{t_k \in \mathcal{E}_j} \left[(u_{em}(t_k) - u_{th}(t_k - \delta\Delta))^2 + (v_{em}(t_k) - v_{th}(t_k - \delta\Delta))^2 \right] \right). \quad (5.3)$$

Here, \mathbb{N} , \mathcal{E}_j and \mathcal{S} represent the set of natural numbers, the set of time indices associated with reconstructed trajectories for the j -th trial, and the index set of all trials under consideration, respectively. It is *essential to incorporate delays* ($\delta\Delta$) into the theoretical curvature terms in the mismatch minimization problem, to take into account the latency present in the sensorimotor feedback loops and to estimate it from the data. We should also note that the discreteness of delay values (in

Strategy	Steering Feedback Law
CATD/MC:	$u_{th} = -\frac{\mu}{\nu_p} \left(\mathbf{z}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{ \mathbf{r} } \right) \right)$ $v_{th} = \frac{\mu}{\nu_p} \left(\mathbf{y}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{ \mathbf{r} } \right) \right)$
CP:	$u_{th} = -\frac{\mu}{\nu_p} \left(\mathbf{y}_p \cdot \frac{\mathbf{r}}{ \mathbf{r} } \right) - \frac{1}{\nu_p \mathbf{r} } \left(\mathbf{z}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{ \mathbf{r} } \right) \right)$ $v_{th} = -\frac{\mu}{\nu_p} \left(\mathbf{z}_p \cdot \frac{\mathbf{r}}{ \mathbf{r} } \right) + \frac{1}{\nu_p \mathbf{r} } \left(\mathbf{y}_p \cdot \left(\dot{\mathbf{r}} \times \frac{\mathbf{r}}{ \mathbf{r} } \right) \right)$

Table 5.1: Theoretically plausible feedback laws for constant absolute target direction (CATD/MC) and classical pursuit (CP).

(5.3)) arises only because of data availability at a finite sample rate (with sampling interval Δ). Clearly, in addition to gathering evidence for a particular pursuit strategy, this approach also yields an estimate of behavioral latency (sensorimotor delay) associated with the pursuit events.

Proposition 5.1. Let $\{a_k\}_{k=0}^N$ and $\{\tilde{a}_k\}_{k=0}^N$ be two finite sequences, and consider the following optimization problem:

$$\underset{\mu > 0, \delta \in \mathbb{N}, \eta \in \mathbb{R}}{\text{Minimize}} \left(\frac{1}{N + 1 - \delta} \sum_{k=\delta}^N \|a(k) - \mu \tilde{a}(k - \delta) - \eta\|^2 \right). \quad (5.4)$$

Then (5.4) can be approximated by the following optimization problem

$$\underset{\delta \in \mathbb{N}}{\text{Maximize}} \text{Corr}(a, \tilde{a}^\delta), \quad (5.5)$$

where \tilde{a}^δ represents a δ -shifted copy of $\{\tilde{a}_k\}$, i.e. $\tilde{a}^\delta(k) = \tilde{a}(k - \delta)$.

Moreover, the optimal values of μ and η are given by

$$\mu = \frac{\text{Cov}(a, \tilde{a}^\delta)}{\text{Var}(\tilde{a}^\delta)} \quad (5.6)$$

$$\eta = \frac{1}{N+1-\delta} \sum_{k=\delta}^N \left(a(k) - \mu \tilde{a}(k-\delta) \right) \quad (5.7)$$

Proof. One can notice that (5.4) can be expressed as

$$\underset{\delta \in \mathbb{N}}{\text{Minimize}} \left(\underset{\mu > 0, \eta \in \mathbb{R}}{\text{Minimize}} \left(\frac{1}{N+1-\delta} \sum_{k=\delta}^N \|a(k) - \mu \tilde{a}(k-\delta) - \eta\|^2 \right) \right), \quad (5.8)$$

and this formulation enables us to solve the optimization problem through a two-step process. Now, for a given δ , we define $a_{avg} = \frac{1}{N+1-\delta} \sum_{k=\delta}^N a(k)$ and $\tilde{a}_{avg} = \frac{1}{N+1-\delta} \sum_{k=0}^{N-\delta} \tilde{a}(k)$. Then, for that particular choice of δ , we have

$$\begin{aligned} & \sum_{k=\delta}^N \|a(k) - \mu \tilde{a}(k-\delta) - \eta\|^2 \\ &= \sum_{k=\delta}^N \left((a(k) - a_{avg}) - \mu (\tilde{a}(k-\delta) - \tilde{a}_{avg}) + (a_{avg} - \mu \tilde{a}_{avg} - \eta) \right)^2 \\ &= \sum_{k=\delta}^N \left(b(k) - \mu \tilde{b}(k-\delta) + (a_{avg} - \mu \tilde{a}_{avg} - \eta) \right)^2 \\ &= \sum_{k=\delta}^N (b^2(k) - 2\mu b(k) \tilde{b}(k-\delta) + \mu^2 \tilde{b}^2(k-\delta)) + \sum_{k=\delta}^N (a_{avg} - \mu \tilde{a}_{avg} - \eta)^2 \\ & \quad + 2 \sum_{k=\delta}^N (b(k) - \mu \tilde{b}(k-\delta)) (a_{avg} - \mu \tilde{a}_{avg} - \eta) \end{aligned} \quad (5.9)$$

where $b(k) \triangleq a(k) - a_{avg}$ and $\tilde{b}(k) \triangleq \tilde{a}(k) - \tilde{a}_{avg}$ ($k \in \{\delta, \dots, N\}$) are two zero mean

sequences. As the third term of (5.9) ceases to zero, one can notice that (5.9) get minimized by choosing

$$\eta = a_{avg} - \mu \tilde{a}_{avg} = \frac{1}{N+1-\delta} \sum_{k=\delta}^N (a(k) - \mu \tilde{a}(k-\delta)) \quad (5.10)$$

$$\mu = \frac{\sum_{k=\delta}^N b(k) \tilde{b}(k-\delta)}{\sum_{k=\delta}^N \tilde{b}^2(k-\delta)} = \frac{\text{Cov}(a, \tilde{a}^\delta)}{\text{Var}(\tilde{a}^\delta)}. \quad (5.11)$$

Replacing μ and η with their optimal values, (5.9) can be expressed as

$$\begin{aligned} & \sum_{k=\delta}^N \left(b^2(k) - 2 \frac{\text{Cov}(a, \tilde{a}^\delta)}{\text{Var}(\tilde{a}^\delta)} b(k) \tilde{b}(k-\delta) + \left(\frac{\text{Cov}(a, \tilde{a}^\delta)}{\text{Var}(\tilde{a}^\delta)} \right)^2 \tilde{b}^2(k-\delta) \right) \\ &= (N+1-\delta) \left(\text{Var}(a) - 2 \frac{\text{Cov}(a, \tilde{a}^\delta)}{\text{Var}(\tilde{a}^\delta)} \text{Cov}(a, \tilde{a}^\delta) + \left(\frac{\text{Cov}(a, \tilde{a}^\delta)}{\text{Var}(\tilde{a}^\delta)} \right)^2 \text{Var}(\tilde{a}^\delta) \right) \\ &= -(N+1-\delta) \text{Var}(a) \left(\frac{(\text{Cov}(a, \tilde{a}^\delta))^2}{\text{Var}(a) \text{Var}(\tilde{a}^\delta)} - 1 \right) \end{aligned} \quad (5.12)$$

as both b and \tilde{b} zero mean sequences. By using the minimum value from the inner optimization (5.12), the problem of our interest can be expressed as

$$\begin{aligned} & \text{Maximize}_{\delta \in \mathbb{N}} \left(\text{Var}(a) \left(\frac{(\text{Cov}(a, \tilde{a}^\delta))^2}{\text{Var}(a) \text{Var}(\tilde{a}^\delta)} - 1 \right) \right) \\ & \Leftrightarrow \text{Maximize}_{\delta \in \mathbb{N}} \text{Var}(a) \left((\text{Corr}(a, \tilde{a}^\delta))^2 - 1 \right). \end{aligned} \quad (5.13)$$

Assuming the delay to be sufficiently small compared to the length of the sequence ($\delta \ll N$), we can ignore the effect of δ on the empirical (sample) variance of a (first factor of the cost). As a consequence, the original optimization problem can be

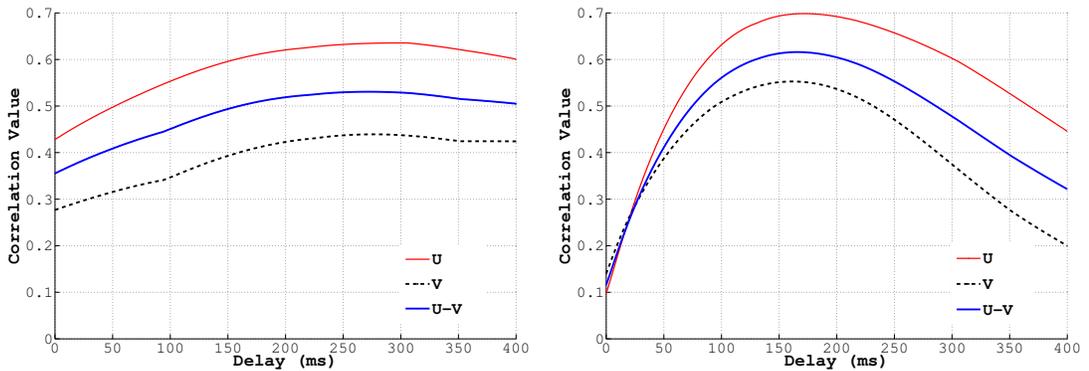
approximated as

$$\boxed{\text{Maximize}_{\delta \in \mathbb{N}} \text{Corr}(a, \tilde{a}^\delta)}. \quad (5.14)$$

□

One can readily recognize that solving the minimization problem (5.3) is computationally demanding. However, this computational complexity can be tackled by approximating it with a correlation maximization problem (as shown via Proposition 5.1). In this alternative approach, we compute the correlation between empirical data (natural curvatures u_{em}, v_{em} stacked in a single array) and the theoretically predicted curvature values (u_{th}, v_{th} stacked in a single array), as a function of delay. Then, the delay which maximizes this correlation provides an estimate for the behavioral latency.

For the bat-bat pursuit events the variation of correlation is shown in Fig 25, and the corresponding values of optimal gain (μ^*) and optimal delay (δ^*) are pro-



(a) Correlation between empirical curvatures and the curvatures obtained from the CATD law. (b) Correlation between empirical curvatures and the curvatures obtained from the CP feedback law.

Figure 25: Variation of correlation between empirical and theoretical curvatures shows dominance of classical pursuit during bat-bat pursuit events.

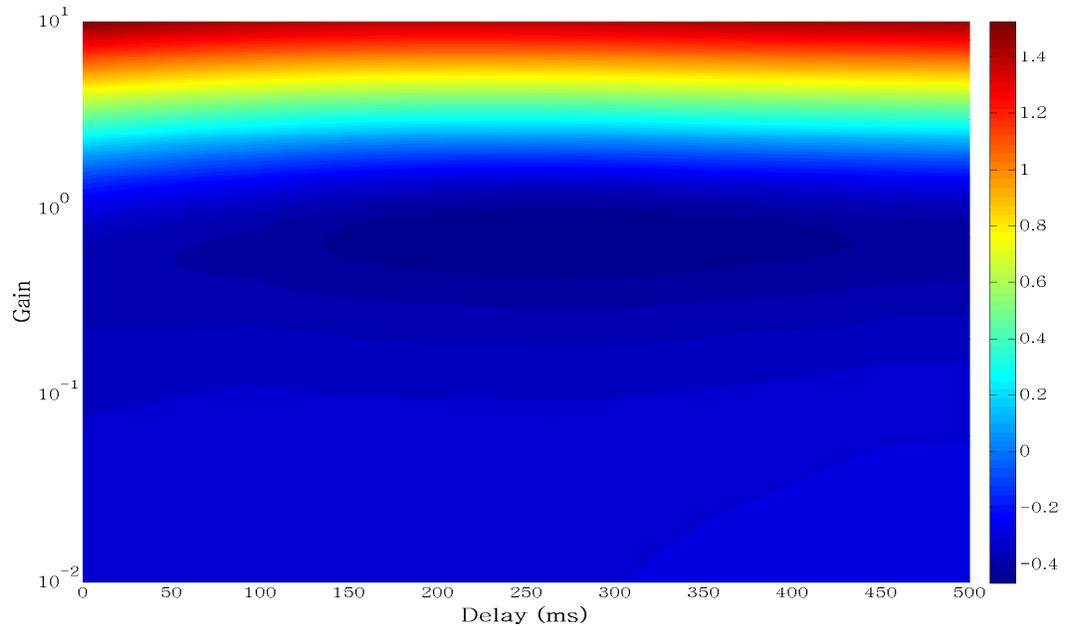


Figure 26: Variation of residual for a CATD feedback law (Bat-Bat pursuit events). The residual values (color-coded) are shown in log 10 scale.

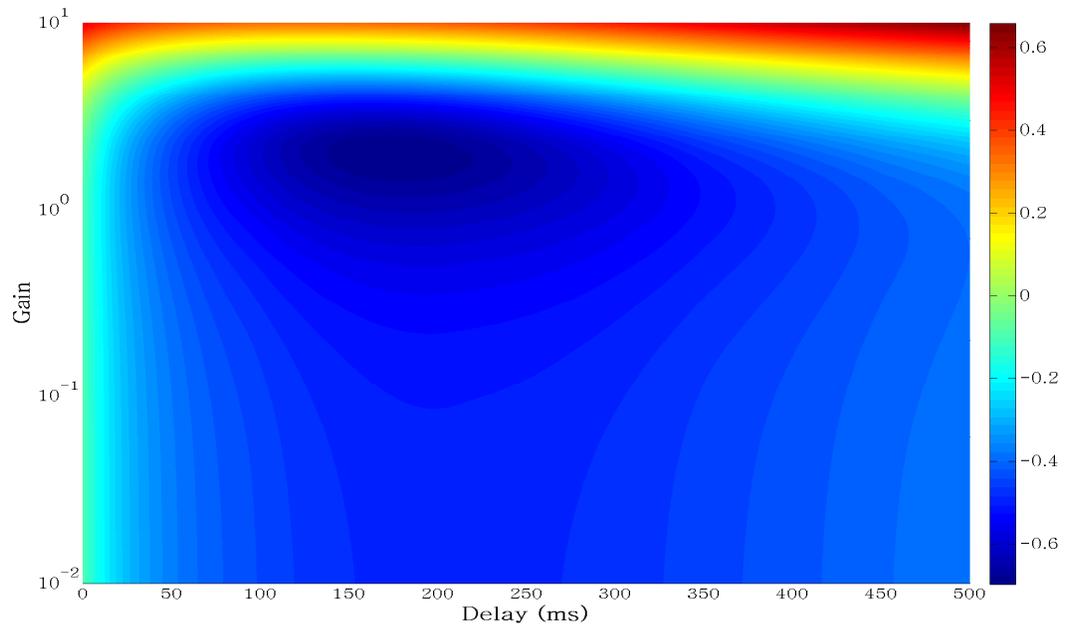


Figure 27: Variation of residual for a CP feedback law (Bat-Bat pursuit events). The residual values (color-coded) are shown in log 10 scale.

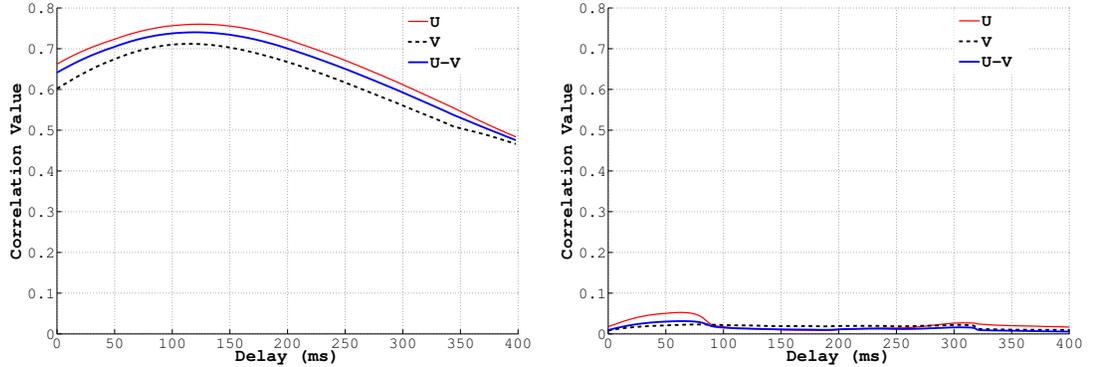
	CATD	CP
Maximum Correlation (ρ_{max})	0.5307	0.6159
Delay (δ^*) [ms]	272.9167	166.6667
Linear Gain (μ_{est})	0.7156	1.9792
λ_1	0.6256	0.3378
λ_2	0.1607	0.0188
σ_1	79.5584	94.7238
σ_2	20.4416	5.2762

Table 5.2: Summary of the statistical analysis of steering control laws for bat-bat pursuit events. λ_1 and λ_2 represent the principal component variances, i.e., eigenvalues of the covariance matrix. σ_1 and σ_2 represent the percentage of total variance explained by principle components.

	CATD	CP
Minimum Value of Mismatch	0.3429	0.2010
Normalized Mismatch	1.0396	0.6216
Linear Gain	0.7165	1.9658
Delay (δ^*) [ms]	277.0833	177.0833

Table 5.3: Summary of the residual analysis of steering control laws for bat-bat pursuit events. The mismatch is normalized by the product of rms values of empirical and theoretical curvatures.

vided in Table 5.2. One can notice that the correlation between the theoretical and empirical values of the curvatures ($u-v$ stacked together) attains maximum (at



(a) Correlation between empirical curvatures (b) Correlation between empirical curvatures and the curvatures obtained from the CATD and the curvatures obtained from the CP feedback law.

Figure 28: Variation of correlation between empirical and theoretical curvatures shows dominance of CATD pursuit strategy during a bat-mantis chase.

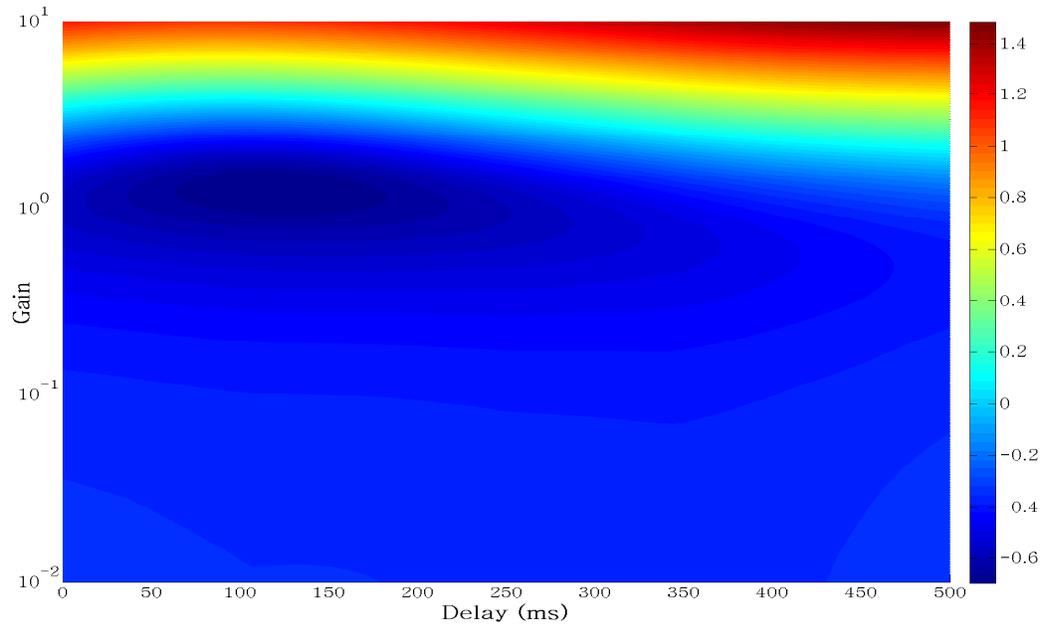


Figure 29: Variation of residual for a CATD feedback law (Bat-Mantis pursuit events). The residual values (color-coded) are shown in log10 scale.

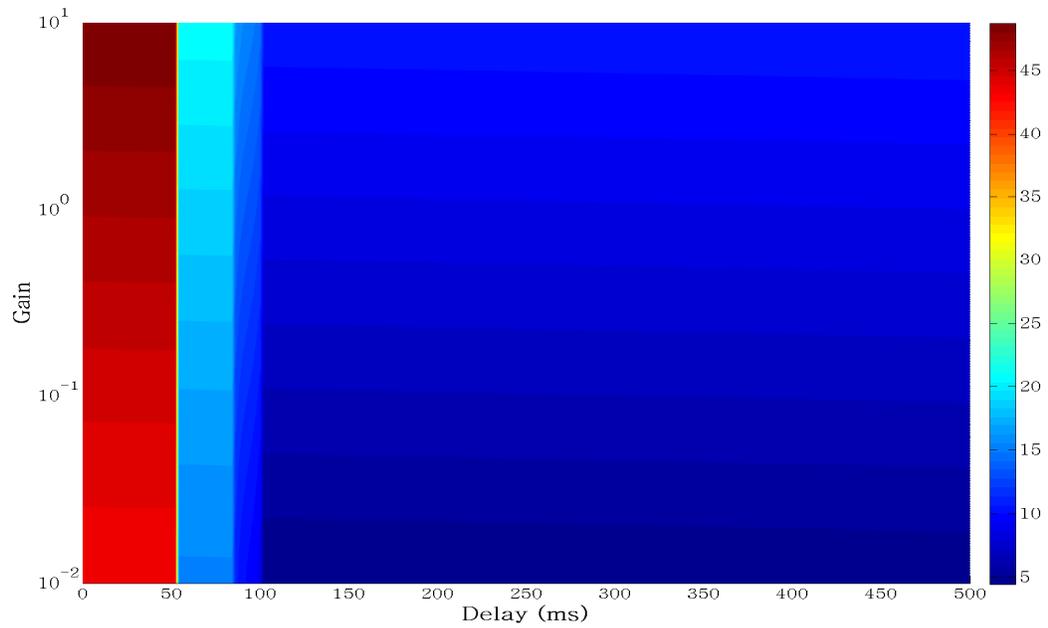


Figure 30: Variation of residual for a CP feedback law (Bat-Mantis pursuit events). The residual values (color-coded) are shown in log10 scale.

0.6159) with the CP feedback law, and the corresponding delay is $166.67ms$. We also perform a principal component analysis to confirm that the data is directional. The results show that the variance of one of the components is very high compared to the other. We also show the variation of the residual (mismatch defined in (5.3)) as a function of gain and delay. From Table 5.3 one can conclude that CP feedback laws yield a better match, and the corresponding gain and delay are similar to the ones obtained through correlation maximization.

On the other hand, we show the variation of correlation for bat-mantis pursuit events in Fig 28, and the corresponding values of optimal gain (μ^*) and optimal delay (δ^*) are mentioned in Table 5.4. It can be noticed from the table that the correlation between the theoretical and empirical values of the curvatures ($u-v$ stacked together) gets maximized (at 0.7403) by choosing the CATD feedback law, and the corresponding delay is $120ms$. A principal component analysis of the curvature data shows that the variance of one of the components is very high compared to the other. In addition to computing the variation of correlation as a function of delay, we also analyze the variation of the residual as a bi-variate function of gain and delay (Fig 29 and Fig 30).

We can observe that the empirical curvature values of the trailing bat in a bat-bat pursuit event are better correlated with a the CP feedback law; while on the other hand the empirical curvature values of the bat in a bat-mantis chase are better correlated with the theoretical feedback law which makes the interaction approach the CATD state. We have also found that the latency associated with the pursuit of a free flying target ($\approx 120ms$) is significantly smaller than the latency associated

	CATD	CP
Maximum Correlation (ρ_{max})	.7403	0.0314
Delay (δ^*) [ms]	120	64
Linear Gain (μ_{est})	1.2457	8.2354×10^{-7}
λ_1	0.5480	374938355.1287
λ_2	0.0586	0.25895
σ_1	90.3439	100
σ_2	9.6561	6.9066×10^{-8}

Table 5.4: Summary of the statistical analysis of steering control laws for bat-mantis pursuit events. λ_1 and λ_2 represent the principal component variances, i.e., eigenvalues of the covariance matrix. σ_1 and σ_2 represent the percentage of total variance explained by principle components.

	CATD	CP
Minimum Value of Mismatch	0.2029	2.8964×10^4
Normalized Mismatch	0.0511	7.6092×10^3
Linear Gain	1.2457	0.0100
Delay (δ^*) [ms]	118	102

Table 5.5: Summary of the residual analysis of steering control laws for bat-mantis pursuit events. The mismatch is normalized by the product of rms values of empirical and theoretical curvatures.

with a competitive prey capture ($\approx 170ms$). This delay can be attributed to the combined effect of neural processing delay and motor action delay.

5.5 Discussion

The results of this study demonstrate that the echolocating bat allows room for flexibility in its flight strategy. It adapts the underlying strategy and feedback mechanism for steering control to the context and goal of the task. Our analysis, built on geometric notions and control-theoretic methods, shows that single bats employ CATD while pursuing insect prey. On the other hand, when tasked to compete for a single food source, a bat does not apply CATD to chase a conspecific. Rather, it uses CP strategy while pursuing a competitor.

This work introduces two important notions, namely following property and convergence of a trajectory, to better understand the individual roles in a dyadic interaction. One distinguishing feature of these notions, similar to the ones introduced by Chiu *et al.* [2008], is that they are completely based on the geometry of flight trajectories, in particular individual positions and velocities; our approach does not consider the patterns of vocalization (for echolocation and warning conspecifics).

Here, we have adopted a two stage approach for analysis of the trajectory data. Besides analyzing the distribution of relevant contrast function (Λ and Γ) values, we have also compared the empirical values of trajectory curvatures to the ones predicted by theoretically well-founded and biologically plausible feedback control laws. While analyzing the histograms of contrast function values, we noticed that the significance of distinction between CP and CATD is higher for bat-bat interactions than for bat-insect interactions. On the other hand, by performing the analysis at the level of steering control law, the dominance of CATD during a bat-insect interaction can be concluded in a much stronger way than the dominance of CP during a bat-bat interaction. This observation emphasizes the indispensability of a two stage approach to achieve a better understanding of the interactions and to evaluate which strategy prevails.

During the trials involving two competing bats, our data shows emergence of a pursuer-pursuee relationship. The approach taken here does not address how this shift to asymmetry arises out of a symmetric setup. It has been known for some time that food-associated vocalization may be responsible for the emergence of such asymmetry (either by attracting or repelling conspecifics). Recent work by Wright

et al. [2014] suggests that this occurs in competitive foraging, when one of the bats (exclusively male) emits a type of social call - frequency-modulated bout (FMB) - to signify territoriality and food claiming. It would be of interest to find signatures in flight steering of such FMB-induced shifts.

Chapter 6: Analysis of Flocking in European Starlings

Collective behavior in animal groups is quite fascinating and ubiquitous in nature, and over the decades had drawn attention of researchers from various fields of science. However, until recently it was very difficult to conduct any quantitative analysis for large groups of animals which are quite common in natural settings. This lack of studies, primarily caused by inadequacy of appropriate motion capture techniques to track common flocks, used to pose a serious problem as an analysis for a small group cannot be generalized for larger groups. The boundary effect is much more dominant for smaller groups.

Equipped with advanced tools from stereometry and statistical analysis, Cavagna *et al.* [2008*a,b*] developed a way to collect three-dimensional position and velocity data for large flocks of starling with number of birds varying from couple of hundreds to couple of thousands. One of their initial contribution was to show that local interactions, the building block for group level flocking behavior, does not depend on metric distance, rather on the topological distance [Ballerini *et al.*, 2008*a*]. Based on the empirical data, they discovered that individual starlings interact with six/seven nearest neighbors (on average), rather than with all neighbors within a fixed radius. A more recent work, by suggesting that the flock maximizes

its robustness to uncertainty by interacting with six or seven neighbors, provides a justification for individual starlings to interact with six/seven nearest neighbors [Young *et al.*, 2013]. Later, by measuring the correlation between velocity fluctuations of different birds, Cavagna *et al.* [2010] have shown that behavioral correlations are scale free, i.e. the behavioral change of an individual affects and is affected by that of all other members of the flock, independent of flock size. In the most recent work from this research group, Attanasi *et al.* [2014] have shown that directional information within a flock propagates with an almost constant speed, and this linear growth of information can be explained by models with wave-like aspects. However, these studies do not provide much insight about the agent-level steering control laws which give rise to flocking behavior.

In this work of ours we attempt to uncover the flight strategies and underlying control laws by analyzing different parameters of motion, namely velocity, speed, curvatures etc. For that purpose we assume each starling to be a point particle, and apply the tools developed in Chapter 3 to extract speed and curvatures from the sampled dataset of observed positions. Then we perform correlation analysis to investigate the feedback mechanism for steering control governing coordinate motion of the flocks. Our analysis also provides estimates of the sensorimotor delay associated with the flocking behavior.

6.1 Experiment Details and Trajectory Reconstruction

This work analyzes the flight data from a series of data collection events conducted by Dr. Andrea Cavagna and his collaborators from the Collective Behaviour in Biological Systems (COBBS) group at the Institute for Complex Systems (IS-CNR), University of Rome “La Sapienza”. These time-sampled flight data were taken from the roof of Palazzo Massimo, Museo Nazionale Romano, in the city center of Rome, in front of one of the major roosting sites used by starlings during winter. Starlings spend the day feeding in the countryside, and before settling on the trees for the night they gather in flocks to perform *aerial display*, an apparently purposeless dance where flocks move and swirl in a remarkable way. Interested readers can refer to the work by Attanasi, A. and Cavagna, A. and Del Castello, L. and Giardina, I. and Jelic, A. and Melillo, S. and Parisi, L. and Shen, E. and Silvestri, E. and Viale, M. [2013] for further details about the experimental setup and the sophisticated algorithm for stereo reconstruction.

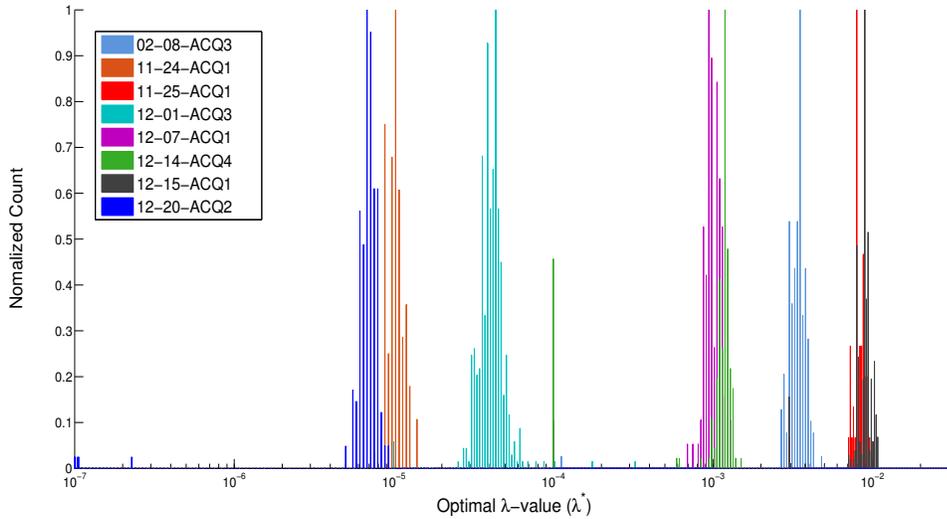


Figure 31: *This figure shows the distribution of optimal regularization parameters (λ^*) for different flocks. The distribution corresponding to a particular flocking event is sharply peaked at a distinct value, and therefore emphasizes the strong dependence of λ^* on the signal-to-noise ratio.*

Our work involves analysis of eight distinct flocking events captured during the winter months of 2011. We begin our analysis by reconstructing the flight trajectories using the smoothing algorithm (Algorithm 1) developed in Chapter 3. In contrast to our earlier work with bat trajectories, here we have performed cross-validation for individual trajectories and the corresponding optimal value of regularization parameter (λ^*) has been used for trajectory reconstruction. Fig 31 shows the distribution of λ^* , and thereby emphasizes that λ^* has a strong dependence on the signal-to-noise ratio. The flocking events that we analyze are quite distinct in nature; while some of them (Fig 32d, Fig 32h) are minimally maneuvering flights, some involve coordinated turning (Fig 32a, Fig 32f, Fig 32g). The following table (Table 6.1) enlists the events under consideration along with the associated details (duration, flock size and frame-rate of data capture).

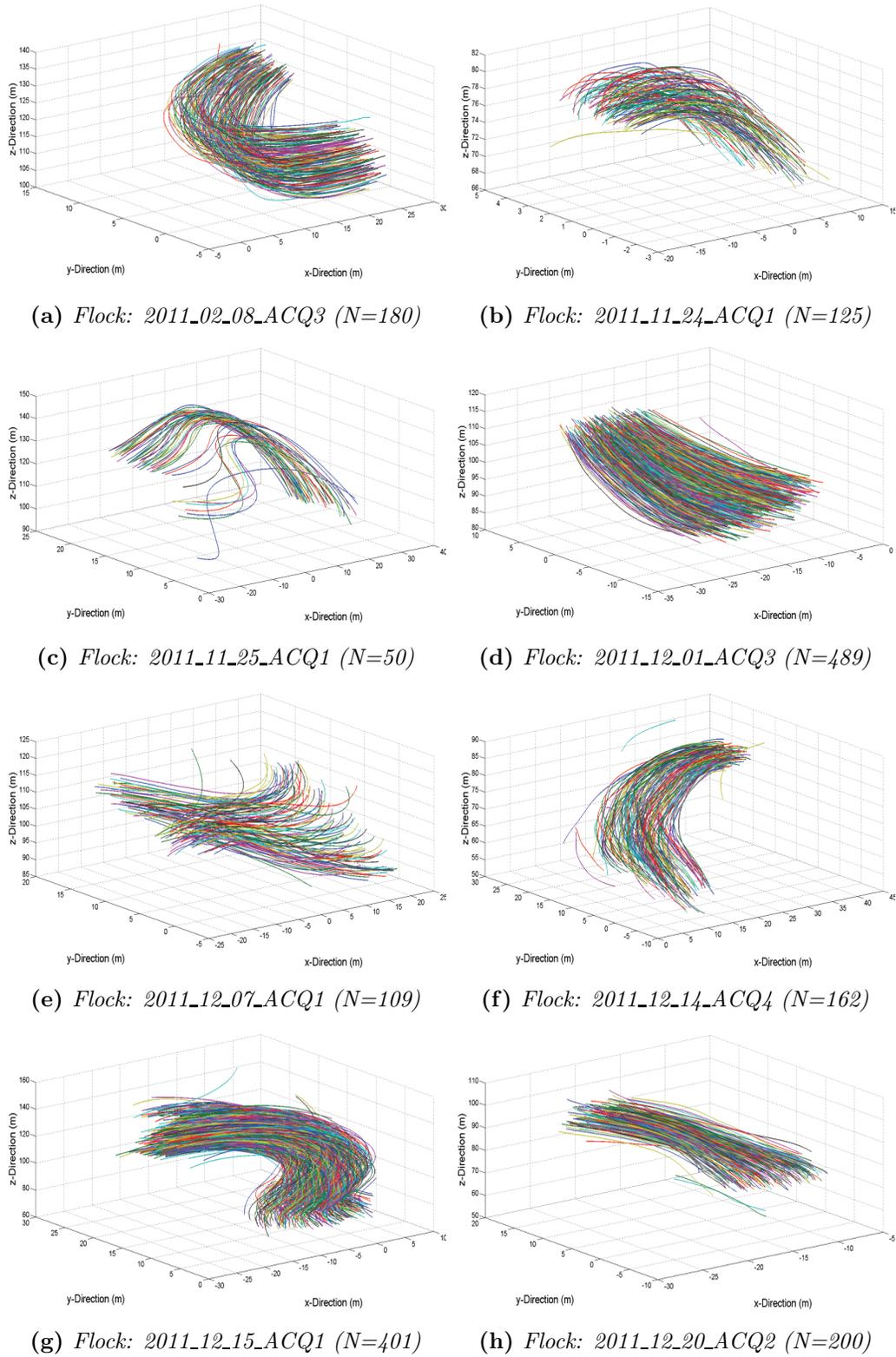


Figure 32: Within the scope of this current work we have analyzed the flight data of eight flocks. This figure shows the reconstructed trajectories for all these flocking events. Here “N” represents the number of birds in a particular flock.

Flocking Event	Flock Size (N)	Duration (seconds)	Data Capture Rate (frames/second)
2011_02_08_ACQ3	180	5.4875	80
2011_11_24_ACQ1	125	1.8176	170
2011_11_25_ACQ1	50	5.6118	170
2011_12_01_ACQ3	489	2.3471	170
2011_12_07_ACQ1	109	3.8824	170
2011_12_14_ACQ4	162	4.1588	170
2011_12_15_ACQ1	401	5.7353	170
2011_12_20_ACQ2	200	1.7588	170

Table 6.1: *Our study analyzes eight particular flocking events. This table enlists the details of the individual events.*

6.2 Analysis of Flight Strategy and Underlying Steering Control

One can easily notice (from Fig 32) that individuals in a starling flock fly in such a way that there is not much variation between the individual directions of motion (at least when they are involved in a coordinated turn or undergoing a minimally maneuvering path). This perception forms the basis for our analysis, and we introduce appropriate quantitative notions to investigate the applicability of this idea.

We begin the analysis by computing the average *cosine* of the angle between the velocity of a focal bird i and velocity of the center of mass of its neighborhood (\mathcal{N}_i). By letting \mathbf{v}_i denote the velocity of the i -th individual, the direction of motion of its neighborhood at time t_k can be defined as

$$\mathbf{x}_{\mathcal{N}_i}(t_k) = \frac{\sum_{j \in \mathcal{N}_i(t_k)} \mathbf{v}_j(t_k)}{\left| \sum_{j \in \mathcal{N}_i(t_k)} \mathbf{v}_j(t_k) \right|}, \quad (6.1)$$

whenever the neighborhood center of mass velocity does not vanishes to zero. Then

the average cosine of the angle (between \mathbf{v}_i and $\mathbf{x}_{\mathcal{N}_i}$) can be computed as

$$C_i = \frac{1}{|\mathcal{E}|} \sum_{k \in \mathcal{E}} \frac{\mathbf{v}_i(t_k)}{|\mathbf{v}_i(t_k)|} \cdot \mathbf{x}_{\mathcal{N}_i}(t_k), \quad (6.2)$$

where \mathcal{E} represents the set of time indices associated with the flight duration. As C_i can be interpreted as a measure of how coherently a bird is moving with respect to its neighbors, C_i is named as the *coherence* for bird i .

This work constructs the neighborhood of a focal bird i by considering its K -nearest neighbors. Then, by definition, \mathcal{N}_i has a cardinality of K , and each node in the underlying *attention graph*¹ has an out-degree of value K . For this particular choice for the neighborhood structure, we evaluate C_i for every member of the flock, and compute the *flock averaged coherence* by averaging it over the flock members. Clearly, this quantity is a property of the whole flock, and it depends only on the neighborhood size (K). Fig 33 shows the variation of flock averaged coherence as a function of the neighborhood size (K).

It can readily be noticed from this set of figures that the flock averaged coherence gets maximized by considering 5-7 nearest neighbors. While some of the flocks (Fig 33c, Fig 33e, Fig 33f) attain the maximum coherence by picking K -values in the range (5-7), for rest of them, the gain/increase in flock averaged coherence becomes insignificant if one goes beyond this range of nearest neighbors. Thus our analysis

¹As the nearest neighborhood relationship is not symmetric, it can be illustrated by a directed graph, wherein an edge exists from node- i to node- j if the individual j is one of the K -nearest neighbors of individual i (i.e., individual i is paying attention to individual j). This graph is called the underlying *attention graph*. Clearly, this graph changes over time, depending on the relative position of the group members.

reconfirms the findings by Ballerini *et al.* [2008a].

Next we focus on investigating the underlying steering control mechanism by comparing empirical values of trajectory curvature with predictions from theoretically justifiable feedback control laws (discussed in detail in Chapter 7). By letting, (u_{em}, v_{em}) denote the empirical curvatures obtained from trajectory smoothing and (u_{th}, v_{th}) denote the curvature values computed using theoretical feedback laws, we formalize this analysis as the following *mismatch minimization* problem

$$\text{Minimize}_{\mu > 0, \delta \in \mathbb{N}} \left(\frac{1}{\sum_{j \in \mathcal{S}} (|\mathcal{E}_j| - \delta)} \sum_{j \in \mathcal{S}} \sum_{t_k \in \mathcal{E}_j} \left[(u_{em}(t_k) - u_{th}(t_k - \delta\Delta))^2 + (v_{em}(t_k) - v_{th}(t_k - \delta\Delta))^2 \right] \right), \quad (6.3)$$

Here, \mathbb{N} , \mathcal{E}_j and \mathcal{S} represent the set of natural numbers, the set of time indices associated with the flight duration of individual j , and the index set of all starlings belonging to a particular flock, respectively. This mismatch minimization problem incorporates delays into the theoretical curvature terms to account for the behavioral latency present in the sensorimotor feedback loops. Moreover this approach also provides an estimate of the sensorimotor delay. We should also note that the discreteness of delay values (in (5.3)) arises only because of data availability at a finite sample rate (with sampling interval Δ). Now, before going into the details of analysis, it is worth mentioning the theoretically plausible feedback laws (although the detailed analysis will be discussed later). By letting ν_i denote the speed of the

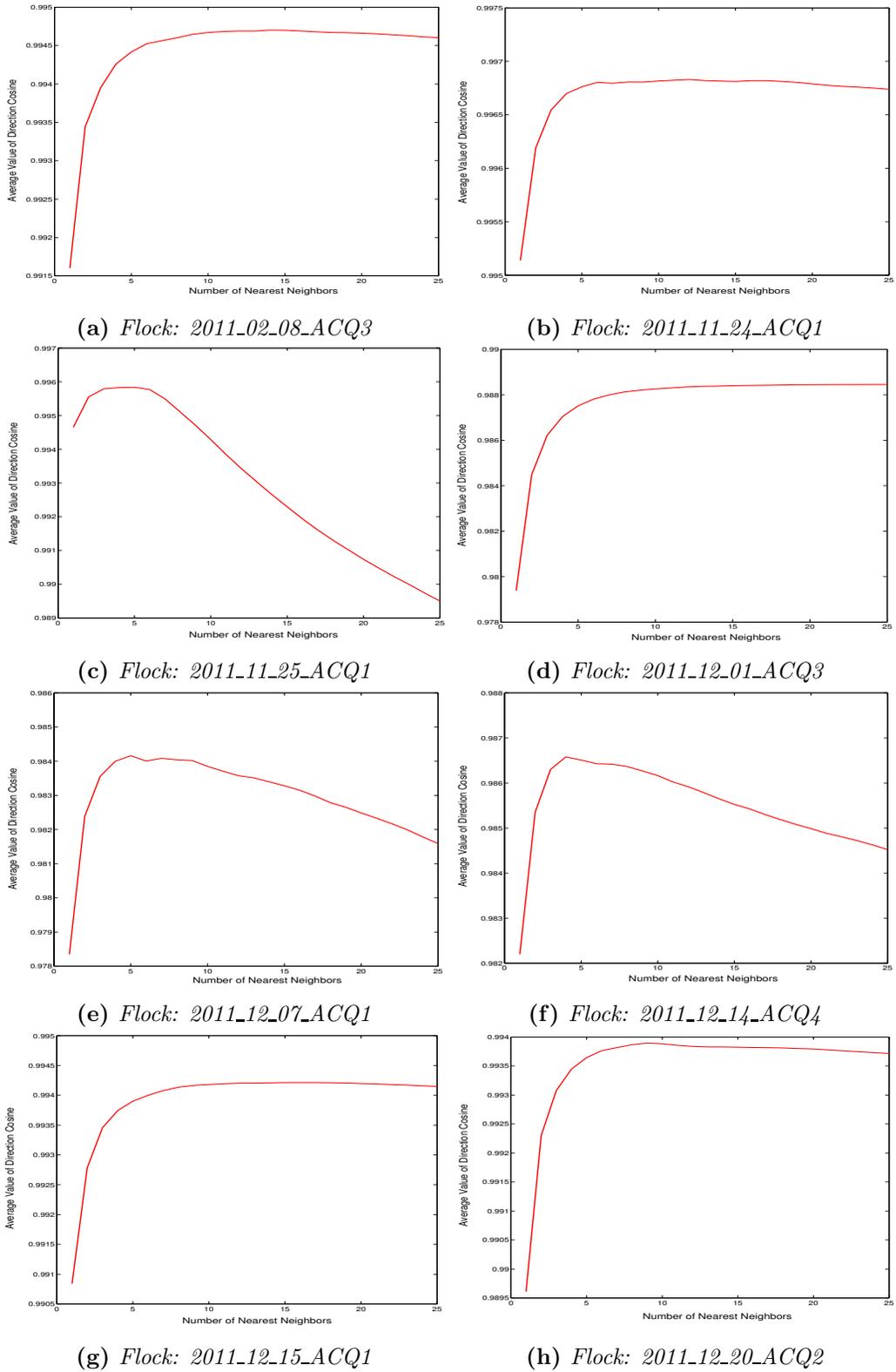
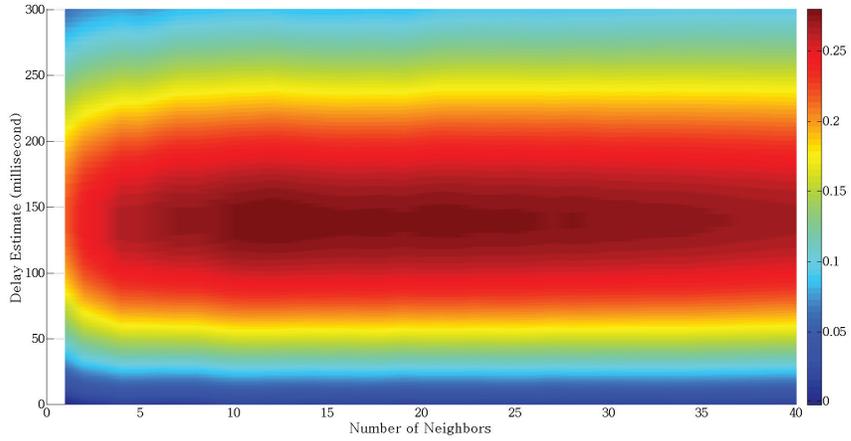
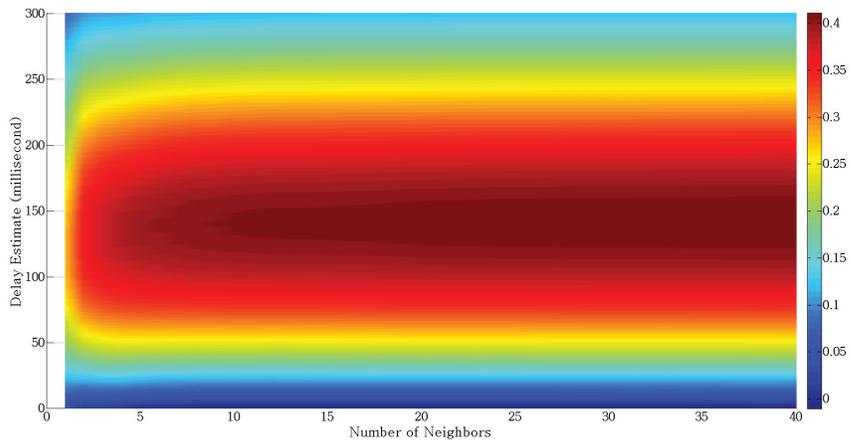


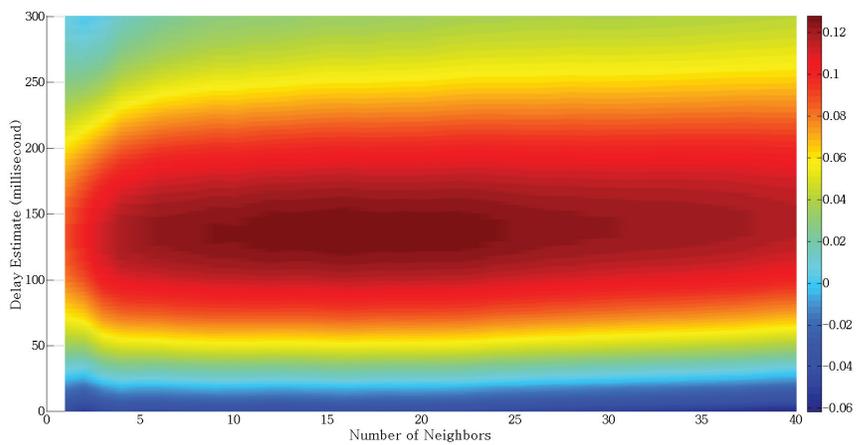
Figure 33: This figure illustrates the variation of flock averaged coherence as a function of the neighborhood size (K), i.e. the number of individuals a focal bird is paying attention to.



(a) Flock: 2011-11-24-ACQ1



(b) Flock: 2011-12-01-ACQ3



(c) Flock: 2011-12-20-ACQ2

Figure 34: This figure, through a heat map, illustrates the variation in correlation between the empirically observed curvature of a focal bird and the curvature values predicted by a theoretically plausible feedback mechanism (6.4), as a bivariate function of neighborhood size and delay.

i -th individual, the feedback law is given by

$$\begin{aligned} u_i &= \mu \left[\frac{\mathbf{x}_{\mathcal{N}_i} \cdot \mathbf{y}_i}{\nu_i} \right] \\ v_i &= \mu \left[\frac{\mathbf{x}_{\mathcal{N}_i} \cdot \mathbf{z}_i}{\nu_i} \right], \end{aligned} \tag{6.4}$$

where $\mathbf{x}_{\mathcal{N}_i}$ denotes the normalized velocity vector of the neighborhood center of mass, and $\mathbf{y}_i, \mathbf{z}_i$ carry their usual meaning of frame vectors associated with the individual. As discussed earlier (in Section 5.4), the computational complexity involved in solving the mismatch minimization problem (6.3) can be avoided by approximating it with an equivalent correlation maximization problem (as shown via Proposition 5.1). In this alternative paradigm, we compute the correlation between empirical data (natural curvatures u_{em}, v_{em} stacked in a single array) and the theoretically predicted curvature values (u_{th}, v_{th} stacked in a single array), as a function of neighborhood size and (sensorimotor) delay. Then, the neighborhood size and delay which maximizes this correlation provides an estimate for the number of nearest neighbors (an individual is paying attention to) and behavioral latency.

Event ID	2011_11_24_ACQ1	2011_12_01_ACQ3	2011_12_20_ACQ2
Max. Correlation	.2797	0.4103	0.1278
Delay [ms]	138.8235	141.1765	135.2941
Neighborhood Size	12	40	16
Linear Gain (μ_{est})	2.4462	4.1689	1.33
λ_1	1271.1725	57592.5342	3505.8485
λ_2	15.2880	462.4669	31.8121
σ_1	98.8116	99.2034	99.1008
σ_2	1.1884	0.7966	0.8992

Table 6.2: Summary of the statistical analysis of steering control laws for starling flocks. λ_1 and λ_2 represent the principal component variances, i.e., eigenvalues of the covariance matrix. σ_1 and σ_2 represent the percentage of total variance explained by the principle components.

The variation of correlation as a bivariate function of neighborhood size and delay is shown in Fig 34, and the associated values of relevant parameters (optimal gain, neighborhood size and delay, along with correlation values) are mentioned in Table 6.2. It can be noticed from the three subfigures that, although the correlation value (between theoretical prediction and empirical observation) is itself not very high, there is an unmistakable consistency in the neighborhood size and delay values which maximize the correlation. These three subfigures shows the correlation variation for the minimally maneuvering flocks (Flock ID: 2011_11_24_ACQ1, Flock ID: 2011_12_01_ACQ3, and Flock ID: 2011_12_20_ACQ2), and hence it can concluded that the proposed feedback mechanism (6.4) plays an important role in governing the steering control for flocking behavior

6.3 Discussion

In this study of ours, we have analyzed the flight data of European starling flocks, and have provided some insight into the underlying flight strategies and steering control mechanism. Here, the attention graph has been constructed by ranking the neighbors according to their distances from the focal bird and then considering the first K -nearest neighbors. However, recent studies by Bhagavatula *et al.* [2011] and Kane & Zamani [2014] have shown that optic flow plays an important role in the context of perception in avian flights. Therefore, it seems reasonable to extend the current work by considering attention graphs constructed by ranking the neighbors according to the magnitude of optic flow at the focal bird's eyes caused by

their motion. Also, it is worth mentioning here that the feedback law, that we have introduced here, is responsible for aligning the individual velocities. A more general form of the control law (although restricted to planar settings and uniformly moving individuals) with three different components for attraction, repulsion and alignment have been proposed earlier by Justh & Krishnaprasad [2002, 2004]. It would be interesting to investigate the applicability of that control law in this context. Another key aspect of this study is that the proposed control law (6.4) carries an wave-like aspect which is necessary to explain a linear growth of information within a flock.

Part III

Synthesis of Collectives

Chapter 7: Flocking through Topological Velocity Alignment (TVA)

Now, drawing inspiration from our analysis of starling murmuration events, we consider the problem of synthesizing a collective motion wherein each of the individuals (birds in a flock, UAVs in swarm) controls its steering action in such a way that their directions of motion remain parallel. It is worth mentioning that earlier works of Justh & Krishnaprasad [2003, 2004] have considered a very similar form of the proposed control law with three components for attraction (while the agents are far away), repulsion (to avoid collision) and velocity alignment. However, the control law introduced in this current study considers only velocity alignment, but extends the scope by considering multiple agents in a three dimensional environment interacting via a state-dependent (nearest neighbors based) attention graph. Moreover, it relaxes the assumption on uniform speed of the collective by allowing the agents non-uniform and time-varying speed profiles. This relaxation plays an important role in the context of applying this control law to a group of heterogeneous agents.

This chapter introduces a strategy, named *topological velocity alignment (TVA)*, wherein each agent aligns its velocity along the direction of motion of its neighborhood center of mass [Halder & Dey, 2015]. After introducing this strategy and the associated steering feedback law, Section 7.2 and Section 7.3 provide a theoretical

analysis of this strategy in two special cases (two-agents, and N -agents interacting in a cyclic way, respectively). Later, in Section 7.4 we propose a discrete-time algorithm to implement the TVA strategy on a group of robotic agents. Finally we conclude this chapter by showing some implementation results in Section 7.5.

7.1 Topological Velocity Alignment (TVA) Strategy

Here we treat the agents (of the collectives of size N) as self steering particles, and use natural Frenet frame equations (Section 1.3.1) to model their dynamics. Therefore, by letting \mathbf{r}_i denote the instantaneous position of the i -th agent, its dynamics can be expressed as

$$\begin{aligned}
 \dot{\mathbf{r}}_i(t) &= \nu_i(t)\mathbf{x}_i(t) \\
 \dot{\mathbf{x}}_i(t) &= \nu_i(t)(u_i(t)\mathbf{y}_i(t) + v_i(t)\mathbf{z}_i(t)) \\
 \dot{\mathbf{y}}_i(t) &= -\nu_i(t)u_i(t)\mathbf{x}_i(t) \\
 \dot{\mathbf{z}}_i(t) &= -\nu_i(t)v_i(t)\mathbf{x}_i(t)
 \end{aligned} \tag{7.1}$$

where \mathbf{x}_i is the unit tangent vector to its trajectory, and $(\mathbf{y}_i, \mathbf{z}_i)$ form an orthonormal basis for the plane perpendicular to \mathbf{x}_i . Moreover the natural curvatures (u_i, v_i) are the steering controls, and the speed (ν_i) is a time function dictated by propulsive/lift/drag mechanisms. It can also be noted that $(\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)$ form a local orthonormal frame which evolves along the length of the trajectory. Furthermore, this model also carries a nice geometric interpretation because (7.1) can alternatively be viewed as a left invariant control system evolving on the Lie group $SE(3)$

of rigid motion in three dimension.

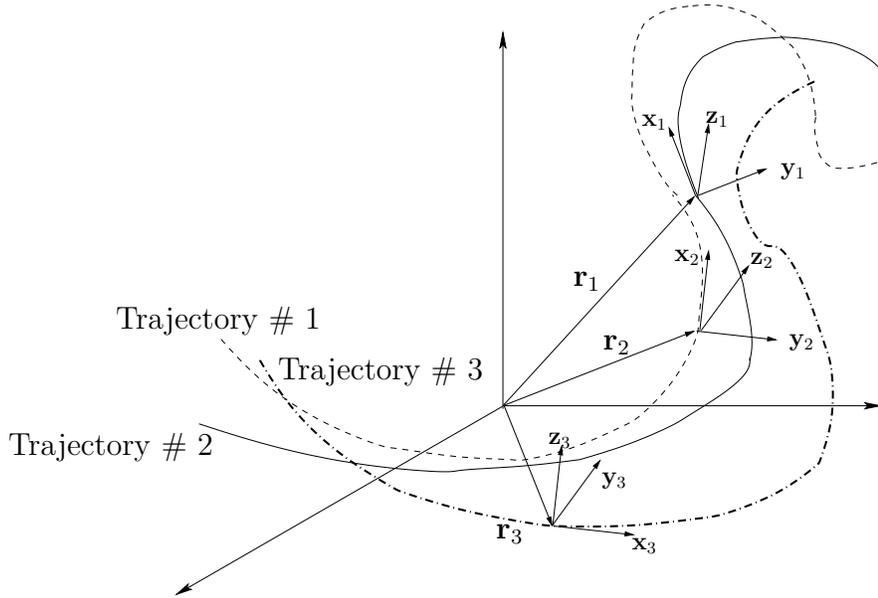


Figure 35: Individual trajectories along with their frame vectors.

By assuming the steering control (u_i, v_i) of an agent- i (or the focal bird in a flock) to be dependent on the motion of its neighbors, we introduce the notion of *topological velocity alignment (TVA)*, wherein each agent moves parallel to the center of mass of its neighborhood. We also derive the steering control law necessary to achieve this goal. We begin by letting \mathcal{N}_i denote the neighborhood of the i -th agent. Then, by assuming identical mass for every agent, the velocity of the neighborhood center of mass (COM) can be expressed as

$$\mathbf{v}_{COM}^i = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \mathbf{v}_j = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} v_j \mathbf{x}_j. \quad (7.2)$$

Moreover, we assume that \mathbf{v}_{COM}^i does not vanish to zero¹. Then, with this assump-

¹It should be noted that \mathbf{v}_{COM}^i becomes zero over a thin set in the underlying state space. As the chance of getting into this thin set is essentially zero, we can overlook this situation for all practical purposes.

tion, we can define the direction of the center of mass motion as

$$\mathbf{x}_{\mathcal{N}_i} = \frac{\mathbf{v}_{COM}^i}{|\mathbf{v}_{COM}^i|} = \frac{\sum_{j \in \mathcal{N}_i} \mathbf{v}_j}{\left| \sum_{j \in \mathcal{N}_i} \mathbf{v}_j \right|}. \quad (7.3)$$

Now, we define the following contrast function associated with the i -th agent

$$\Theta_i = \frac{1}{2} (\mathbf{x}_{\mathcal{N}_i} - \mathbf{x}_i) \cdot (\mathbf{x}_{\mathcal{N}_i} - \mathbf{x}_i), \quad (7.4)$$

and use it as a quantitative measure of the difference between heading of the i -th agent and direction of the center of mass motion for its neighborhood \mathcal{N}_i . If the i -th agent moves in the same direction as the center of mass of its neighborhood \mathcal{N}_i , then the contrast function assumes its minimum value, i.e. $\Theta_i = 0$. On the other hand, it assumes the maximum value ($\Theta_i = 2$) whenever the i -th agent moves in a direction opposite to that of the center of mass motion. And, if the i -th agent moves perpendicular with respect to direction of motion of the center of mass of its neighborhood, then we have $\Theta_i = 1$. Fig 36 shows the variation of Θ_i as a function of the angle between heading of the i -th agent (\mathbf{x}_i) and the direction of the center of mass motion for its neighborhood ($\mathbf{x}_{\mathcal{N}_i}$). As we can observe, Θ_i increases monotonically with increase in the angle between \mathbf{x}_i and $\mathbf{x}_{\mathcal{N}_i}$. Alternatively, it can also be expressed as

$$\Theta_i = 1 - \mathbf{x}_i \cdot \mathbf{x}_{\mathcal{N}_i}, \quad (7.5)$$

as both \mathbf{x}_i and $\mathbf{x}_{\mathcal{N}_i}$ are unit vectors. $\mathbf{x}_i \cdot \mathbf{x}_{\mathcal{N}_i}$, being a dot product of two unit vectors,

essentially represents the *cosine* of the angle between them, and gets maximized when they are aligned in the same direction. Thus, $\Theta_i(t)$ can be interpreted as a quantitative measure of departure from our goal of achieving alignment between heading of the i -th agent ($\mathbf{x}_i(t)$) and the direction of motion of its neighborhood center of mass ($\mathbf{x}_{\mathcal{N}_i}(t)$).

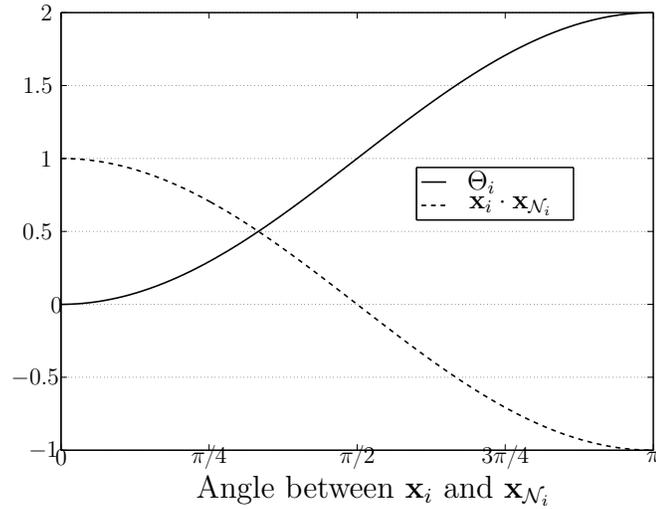


Figure 36: Variation of the contrast function (Θ_i) with change in angle between \mathbf{x}_i and $\mathbf{x}_{\mathcal{N}_i}$.

Assuming a well-defined $\mathbf{x}_{\mathcal{N}_i}$, we propose a steering control law of the form

$$\begin{aligned} u_i(t) &= \mu \left[\frac{\mathbf{x}_{\mathcal{N}_i}(t) \cdot \mathbf{y}_i(t)}{\nu_i(t)} \right] \\ v_i(t) &= \mu \left[\frac{\mathbf{x}_{\mathcal{N}_i}(t) \cdot \mathbf{z}_i(t)}{\nu_i(t)} \right], \end{aligned} \tag{7.6}$$

where $\mu > 0$ denotes a feedback gain. A physical intuition for the steering control law (7.6) can be obtained by shifting focus to the corresponding feedback law for lateral acceleration. With this particular choice of control laws, the lateral acceleration can

be expressed as

$$\begin{aligned}
\mathbf{a}_i^{lat}(t) &= \nu_i(t)\dot{\mathbf{x}}_i(t) \\
&= \mu\nu_i(t)[\mathbf{x}_{N_i}(t) \cdot \mathbf{y}_i(t)]\mathbf{y}_i(t) + \mu\nu_i(t)[\mathbf{x}_{N_i}(t) \cdot \mathbf{z}_i(t)]\mathbf{z}_i(t) \\
&= \mu\nu_i(t)\left(\mathbf{x}_{N_i}(t) - [\mathbf{x}_{N_i}(t) \cdot \mathbf{x}_i(t)]\mathbf{x}_i(t)\right).
\end{aligned} \tag{7.7}$$

From (7.7) it is quite apparent that the lateral acceleration is proportional to the difference between direction of motion of the center of mass and the component of its own tangent vector along the desired direction.

Remark 7.1. *Although there has been a long history of control algorithms for flocking, almost every model of collective motion (Reynolds [1987]; Vicsek et al. [1995]) predicts diffusive transport of information. But, contrary to the existing models, recent findings by Attanasi et al. [2014] from starling flocks suggest that directional information within a flock propagates with an almost constant speed, and this linear growth of information can be explained by models with wave-like aspects. Our proposed strategy, i.e. topological velocity alignment (TVA), conforms to this criterion and can explain how information about local neighbors can influence the agents in a flock to align their headings in a single common direction.*

7.2 TVA Strategy for a Planar 2-agent System

Before going into the analysis of an N -agent system we consider a very special case for a 2-agent system wherein the motion is restricted onto a two dimensional

Cartesian plane. As we are considering a 2-agent system the neighborhood of a particular agent comprises of the other agent, and hence the speed of neighborhood center of mass never vanishes to zero (due to regularity of the model).

7.2.1 State Space and Its Reduction onto Shape Space

By exploiting natural Frenet frame equations restricted to a planar setting, the dynamics of the two agent system can be expressed as

$$\begin{aligned}
 \dot{\mathbf{r}}_i(t) &= \nu_i(t)\mathbf{x}_i(t) \\
 \dot{\mathbf{x}}_i(t) &= \nu_i(t)u_i(t)\mathbf{y}_i(t) \\
 \dot{\mathbf{y}}_i(t) &= -\nu_i(t)u_i(t)\mathbf{x}_i(t) \quad i \in \{1, 2\}
 \end{aligned} \tag{7.8}$$

where \mathbf{r}_i and \mathbf{x}_i denote the position and normalized velocity of the i -th agent, respectively. \mathbf{y}_i is the unit frame vector normal to \mathbf{x}_i (uniqueness is guaranteed by defining \mathbf{y}_i as the orthogonal rotation of \mathbf{x}_i in the counter-clockwise direction). We also assume $\mathbf{r}_1 \neq \mathbf{r}_2$ to avoid singularity of the shape variables (to be introduced later). Moreover, a planar equivalent of (7.6) is chosen as the underlying feedback law for steering control.

Alternatively, by packing $\mathbf{r}_i, \mathbf{x}_i, \mathbf{y}_i$ inside a matrix $g_i \in SE(2)$, the special Euclidean group of rigid motions in a plane, the state space for the 2-agent system (7.8) can be defined as

$$\mathcal{M}_{state} = \left\{ g_1, g_2 \in SE(2) \times SE(2) \mid g_1 e_3 \neq g_2 e_3 \right\} \tag{7.9}$$

where $e_3 = [0 \ 0 \ 1]^T$ and $g_i = \begin{bmatrix} \mathbf{x}_i & \mathbf{y}_i & \mathbf{r}_i \\ 0 & 0 & 1 \end{bmatrix}$. In terms of the lie-group formulation the system dynamics can be represented as

$$\dot{g}_i = g_i \xi_i(u_i) = g_i (A_1 u_i \nu_i + A_2 \nu_i) \quad (7.10)$$

where $A_1 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ and $A_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ are two linearly independent elements of $\mathfrak{se}(2)$, the Lie algebra of $SE(2)$. Moreover A_1 and A_2 can generate the lie-algebra under bracketing. As we are interested in steering laws which leave our system dynamics invariant under rigid motion, we can formulate a reduction to the *shape space*, a quotient manifold $\mathcal{M}_{state}/SE(2)$ of relative positions and velocities of the agents. By defining $g \in SE(2)$ as

$$g = g_1^{-1} g_2 = \begin{bmatrix} \mathbf{x}_1 \cdot \mathbf{x}_2 & \mathbf{x}_1 \cdot \mathbf{y}_2 & \mathbf{x}_1 \cdot \mathbf{r} \\ \mathbf{y}_1 \cdot \mathbf{x}_2 & \mathbf{y}_1 \cdot \mathbf{y}_2 & \mathbf{y}_1 \cdot \mathbf{r} \\ 0 & 0 & 1 \end{bmatrix}, \quad (7.11)$$

the shape space for the planar two-agent system can be defined as

$$\mathcal{M}_{shape} = \mathcal{M}_{state}/SE(2) = \left\{ g \in SE(2) \mid g_{13}^2 + g_{23}^2 \neq 0 \right\}, \quad (7.12)$$

where $\mathbf{r} \triangleq \mathbf{r}_2 - \mathbf{r}_1$ denotes the baseline vector. Moreover, g assumes a left-invariant

dynamics on $SE(2)$ as

$$\dot{g} = g\xi \quad (7.13)$$

where $\xi = \xi_2(u_2) - g^{-1}\xi_1(u_1)g \in \mathfrak{se}(2)$, and the proposed control law (7.6) depends only on the shape variable g as

$$u_i = \mu \begin{bmatrix} g_{\bar{i}i} \\ v_i \end{bmatrix}, \quad i, \bar{i} \in \{1, 2\}, \quad i \neq \bar{i}. \quad (7.14)$$

Therefore it can be concluded that the reduced dynamics (7.13) evolves on the shape space \mathcal{M}_{shape} [Justh & Krishnaprasad, 2004].

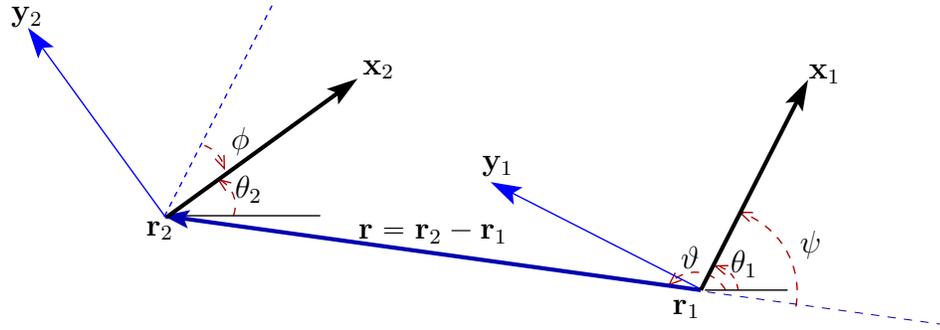


Figure 37: Illustration of scalar shape variables (ρ, ψ, ϕ) used to parametrize the shape space \mathcal{M}_{shape} .

7.2.2 Shape Dynamics

Now we introduce some geometrically meaningful scalar variables to parametrize the shape space. By identifying punctured \mathbb{R}^2 with the punctured complex plane, we define

$$\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1 = \rho e^{i\vartheta}. \quad (7.15)$$

Moreover the unit vectors \mathbf{x}_1 and \mathbf{x}_2 are represented as

$$\mathbf{x}_1 = e^{i\theta_1}, \quad \mathbf{x}_2 = e^{i\theta_2}.$$

These scalar shape variables are illustrated in Fig 37. Now we introduce ψ and ϕ to represent the relative orientation of the velocity vectors. $\psi = \pi - \vartheta + \theta_1$ represents the relative orientation of \mathbf{x}_1 with respect to the baseline vector \mathbf{r} , and $\phi = \theta_1 - \theta_2$ represent the mismatch in velocity direction. From (7.11) one can notice that $g \in \mathcal{M}_{shape}$ can be represented in terms of the scalar shape variables as

$$g = \begin{bmatrix} \cos \phi & \sin \phi & -\rho \cos \psi \\ -\sin \phi & \cos \phi & \rho \sin \psi \\ 0 & 0 & 1 \end{bmatrix}. \quad (7.16)$$

Differentiating both sides, (7.15) yields

$$\begin{aligned} & \dot{\rho} \cos \vartheta - \rho \dot{\vartheta} \sin \vartheta + i(\dot{\rho} \sin \vartheta + \rho \dot{\vartheta} \cos \vartheta) \\ &= \dot{\mathbf{r}} \\ &= \nu_2 \mathbf{x}_2 - \nu_1 \mathbf{x}_1 \\ &= \nu_2 (\cos \theta_2 + i \sin \theta_2) - \nu_1 (\cos \theta_1 + i \sin \theta_1), \end{aligned} \quad (7.17)$$

and then by equating the real and imaginary parts of (7.17) we have

$$\dot{\rho} \cos \vartheta - \rho \dot{\vartheta} \sin \vartheta = \nu_2 \cos \theta_2 - \nu_1 \cos \theta_1 \quad (7.18)$$

$$\dot{\rho} \sin \vartheta + \rho \dot{\vartheta} \cos \vartheta = \nu_2 \sin \theta_2 - \nu_1 \sin \theta_1. \quad (7.19)$$

Solving (7.18) and (7.19) we obtain

$$\begin{aligned} \dot{\rho} &= \nu_2 \cos(\vartheta - \theta_2) - \nu_1 \cos(\vartheta - \theta_1) \\ &= \nu_1 \cos \psi - \nu_2 \cos(\psi - \phi), \end{aligned} \quad (7.20)$$

and

$$\begin{aligned} \rho \dot{\vartheta} &= \nu_2 \sin(\theta_2 - \vartheta) - \nu_1 \sin(\theta_1 - \vartheta) \\ &= \nu_1 \sin \psi - \nu_2 \sin(\psi - \phi). \end{aligned} \quad (7.21)$$

On the other hand, \mathbf{y}_k is obtained by a 90° rotation of \mathbf{x}_k in the counter clockwise direction, i.e. $\mathbf{y}_k = e^{i\frac{\pi}{2}} \mathbf{x}_k$. Therefore by using the fact that $\dot{\mathbf{x}}_k = ie^{i\theta_k} \dot{\theta}_k = e^{i(\theta_k + \frac{\pi}{2})} \dot{\theta}_k$, the dynamics for the scalar variables θ_1 and θ_2 can be represented as

$$\begin{aligned} \dot{\theta}_1 &= \nu_1 u_1 \\ \dot{\theta}_2 &= \nu_2 u_2, \end{aligned} \quad (7.22)$$

and hence the associated shape dynamics on \mathcal{M}_{shape} are given by

$$\begin{aligned}
\dot{\rho} &= \nu_1 \cos \psi - \nu_2 \cos(\psi - \phi) \\
\dot{\psi} &= \nu_1 u_1 - \frac{1}{\rho} [\nu_1 \sin \psi - \nu_2 \sin(\psi - \phi)] \\
\dot{\phi} &= \nu_1 u_1 - \nu_2 u_2.
\end{aligned} \tag{7.23}$$

7.2.3 Analysis of TVA Feedback Law

Here we consider a particular context of the two-agent planar system wherein each agent employs the strategy for topological velocity alignment (TVA), i.e. each agent keeps moving in the same direction as the other. In terms of the original state variables the contrast functions take the form

$$\Theta_i = \frac{1}{2} (\mathbf{x}_i - \mathbf{x}_i) \cdot (\mathbf{x}_i - \mathbf{x}_i), \quad i \in \{1, 2\} \tag{7.24}$$

and the i -th agent is declared to attain the TVA strategy if $\Theta_i = 0$. From (7.24) one can notice equality of the contrast functions for both agents, and hence we define a common contrast function $\Theta = \Theta_1 = \Theta_2$. Noting $\mathbf{x}_1 \cdot \mathbf{x}_2 = \cos \phi$, the common contrast function can be represented in terms of scalar shape variables as

$$\begin{aligned}
\Theta &= \frac{1}{2} (\mathbf{x}_2 - \mathbf{x}_1) \cdot (\mathbf{x}_2 - \mathbf{x}_1) \\
&= 1 - \mathbf{x}_1 \cdot \mathbf{x}_2 \\
&= 1 - \cos \phi,
\end{aligned} \tag{7.25}$$

and hence we have

$$\Theta = 0 \Leftrightarrow \phi = 0. \quad (7.26)$$

Therefore for a two-agent planar system wherein each agent employs the strategy for topological velocity alignment (TVA), we define the 2-dimensional topological velocity alignment manifold $\mathcal{M}_{TVA} \subset \mathcal{M}_{shape}$ as

$$\mathcal{M}_{TVA} = \left\{ \rho, \psi, \phi \in \mathcal{M}_{shape} \mid \phi = 0 \right\}. \quad (7.27)$$

A steering control law designed to attain the TVA strategy has been proposed (7.6) in earlier sections of this chapter. Moreover from (7.14) one can observe that this feedback law can be expressed in terms of shape variables, taking the form

$$\begin{aligned} u_1 &= - \left(\frac{\mu}{\nu_1} \right) \sin \phi \\ u_2 &= \left(\frac{\mu}{\nu_2} \right) \sin \phi. \end{aligned} \quad (7.28)$$

From (7.26) we can observe that once the TVA strategy has been attained, i.e. $\Theta = 0$, the steering control becomes identically zero, and as a consequence the mismatch in velocity direction remains identically zero (7.23). Now we will formally introduce the notion of *invariance*.

Definition 7.1 (Invariant Manifold). *A manifold \mathcal{M} is said to be invariant under the flow of a vector field X on \mathcal{M} if for any $m \in \mathcal{M}$, $F_t(m) \in \mathcal{M}$ for small $t > 0$, where $F_t(\cdot)$ is the flow of X . One can show that this condition is equivalent to X being tangent to the manifold to \mathcal{M} .*

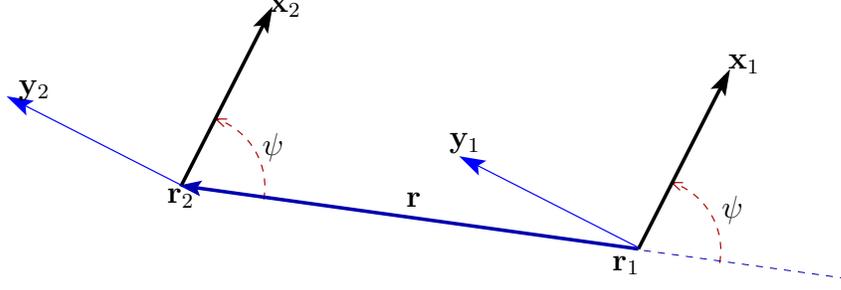


Figure 38: Illustration of topological velocity alignment (TVA) strategy for a 2-agent system.

If both agents employ a steering control of the form (7.28), the closed loop dynamics for a two-agent planar system can be represented as

$$\begin{aligned}
 \dot{\rho} &= \nu_1 \cos \psi - \nu_2 \cos(\psi - \phi) \\
 \dot{\psi} &= -\mu \sin \phi - \frac{1}{\rho} [\nu_1 \sin \psi - \nu_2 \sin(\psi - \phi)] \\
 \dot{\phi} &= -2\mu \sin \phi.
 \end{aligned} \tag{7.29}$$

We should note that prohibition on collocation, i.e. $\rho > 0$, is not enforced by these dynamics.

Proposition 7.1. *The topological velocity alignment manifold $\mathcal{M}_{TVA} \subset \mathcal{M}_{shape}$ is invariant under the closed loop shape dynamics (7.29). Moreover if $\gamma(t) \in \mathcal{M}_{shape}$ is a trajectory of (7.29) which does not have a finite escape time, and $\Theta(0) \neq 2$, then*

$$\Theta(t) \rightarrow 0 \quad \text{as } t \rightarrow \infty, \tag{7.30}$$

i.e. $\gamma(t)$ converges asymptotically to \mathcal{M}_{TVA} .

Proof. From (7.25) and (7.29) we have

$$\dot{\Theta} = \dot{\phi} \sin \phi = -2\mu \sin^2 \phi = -2\mu\Theta(2 - \Theta), \quad (7.31)$$

and hence $\mathcal{M}_{FL} \subset \mathcal{M}_{shape}$ is an invariant manifold under the shape dynamics. In fact, from (7.31) it can be concluded that $\Theta(0) = 2 \Rightarrow \Theta(t) = 2 \quad \forall t \geq 0$.

By assuming $\Theta(0) \neq 0, 2$, we have

$$\Theta(t) = \frac{2e^{-4\mu t}}{K + e^{-4\mu t}}$$

where the constant K is defined as $K = \frac{2}{\Theta(0)} - 1$. Since $e^{-4\mu t} \rightarrow 0$ as $t \rightarrow \infty$, we have $\Theta(t) \rightarrow 0$ as $t \rightarrow \infty$. □

We can formulate the restricted dynamics on the flocking manifold \mathcal{M}_{TVA} by substituting $\phi = 0$ into (7.29) to obtain

$$\begin{aligned} \dot{\rho} &= (\nu_1 - \nu_2) \cos \psi \\ \dot{\psi} &= -\frac{1}{\rho}(\nu_1 - \nu_2) \sin \psi. \end{aligned} \quad (7.32)$$

Now we define $f = \nu_1 - \nu_2$. By assuming f to be non-zero and differentiable

(7.32) can be alternatively represented as

$$\begin{aligned}
\ddot{\rho} &= \dot{f} \cos \psi - f \dot{\psi} \sin \psi \\
&= \dot{f} \cos \psi + \frac{1}{\rho} f^2 \sin^2 \psi \\
&= \left(\frac{\dot{f}}{f} \right) \dot{\rho} + \frac{1}{\rho} (f^2 - \dot{\rho}^2).
\end{aligned} \tag{7.33}$$

Remark 7.2. *If f is a non-zero constant then we can show that a Lagrangian function*

$$L_{TVA} = \frac{1}{2} \rho^2 (\dot{\rho}^2 + f^2)$$

has (7.33) as its Euler-Lagrange equation. Hence it can be interpreted as a spring-mass system with a negative spring constant.

By assuming $\nu_1 - \nu_2 > 0$ we define, $\tau = \int_0^t (\nu_1(\sigma) - \nu_2(\sigma)) d\sigma$ to introduce a non-uniform time-scaling. As a result the restricted dynamics (7.32) can alternatively be represented as

$$\begin{aligned}
\frac{d\rho}{d\tau} &= \cos \psi \\
\frac{d\psi}{d\tau} &= -\frac{1}{\rho} \sin \psi.
\end{aligned} \tag{7.34}$$

Fig 39 shows the phase portrait of the restricted dynamics with a non-uniform time scaling. From this figure one can notice that the region $\{\rho > 0, \pi > \psi > 0\}$ (or similarly $\{\rho > 0, 0 > \psi > -\pi\}$) is closed under the restricted dynamics (7.34), i.e. no trajectory can enter or leave the region. One can notice that the trajectories of (7.32) and (7.34) are essentially the same on the phase plane, they differ only on the speed of system evolution along any particular trajectory. Hence it can be

concluded that the trajectories of (7.32) are also closed within $\{\rho > 0, \pi > \psi > 0\}$ (or similarly within $\{\rho > 0, 0 > \psi > -\pi\}$).

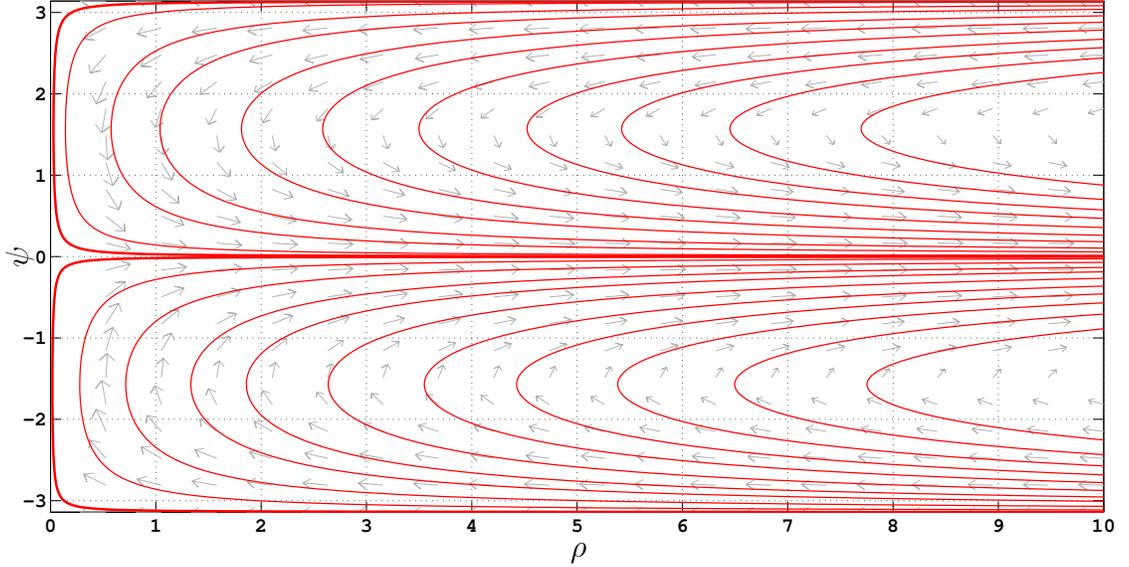


Figure 39: Phase portraits for the restricted dynamics (7.34).

Alternatively, by assuming $\nu_1 - \nu_2 \neq 0$, the evolution of a phase plane trajectory can be represented as

$$\frac{d\rho}{d\psi} = -\frac{\rho \cos \psi}{\sin \psi}, \quad (7.35)$$

and through integration with appropriate initial condition (7.35) yields

$$\rho(t) \sin \psi(t) = \rho(t_0) \sin \psi(t_0). \quad (7.36)$$

From (7.36) one can notice that the phase plane trajectories are level sets of “ $\rho \sin \psi$ ” and hence it is clear that the region $\{\rho > 0, \pi > \psi > 0\}$ (or similarly $\{\rho > 0, 0 > \psi > -\pi\}$) is closed under the restricted dynamics (7.32).

7.2.4 Extension to a Three Dimensional Setting

The analysis of topological velocity alignment for a 2-agent system has a natural extension from planar to a three-dimensional setting, with the underlying state space being

$$\mathcal{M}_{state} = SE(3) \times SE(3). \quad (7.37)$$

For a 2-agent system, the neighborhood of both the agents contains only the other one ($\mathcal{N}_1 = \{2\}$ and $\mathcal{N}_2 = \{1\}$), and as a consequence we have $\Theta_1 \equiv \Theta_2$, because

$$\begin{aligned} \Theta_1(t) &= \frac{1}{2} (\mathbf{x}_{\mathcal{N}_1}(t) - \mathbf{x}_1(t)) \cdot (\mathbf{x}_{\mathcal{N}_1}(t) - \mathbf{x}_1(t)) \\ &= \frac{1}{2} (\mathbf{x}_2(t) - \mathbf{x}_1(t)) \cdot (\mathbf{x}_2(t) - \mathbf{x}_1(t)) \end{aligned} \quad (7.38)$$

$$\begin{aligned} \text{and, } \Theta_2(t) &= \frac{1}{2} (\mathbf{x}_{\mathcal{N}_2}(t) - \mathbf{x}_2(t)) \cdot (\mathbf{x}_{\mathcal{N}_2}(t) - \mathbf{x}_2(t)) \\ &= \frac{1}{2} (\mathbf{x}_1(t) - \mathbf{x}_2(t)) \cdot (\mathbf{x}_1(t) - \mathbf{x}_2(t)). \end{aligned} \quad (7.39)$$

By choosing the steering the control law as the one prescribed in (7.6), we have

$$\begin{aligned} \dot{\Theta}_1 &= (\mathbf{x}_2 - \mathbf{x}_1) \cdot (\dot{\mathbf{x}}_2 - \dot{\mathbf{x}}_1) \\ &= -\nu_2 (u_2 \mathbf{y}_2 \cdot \mathbf{x}_1 + v_2 \mathbf{z}_2 \cdot \mathbf{x}_1) - \nu_1 (u_1 \mathbf{y}_1 \cdot \mathbf{x}_2 + v_1 \mathbf{z}_1 \cdot \mathbf{x}_2) \\ &= -\mu ((\mathbf{y}_2 \cdot \mathbf{x}_1)^2 + (\mathbf{z}_2 \cdot \mathbf{x}_1)^2) - \mu ((\mathbf{y}_1 \cdot \mathbf{x}_2)^2 + (\mathbf{z}_1 \cdot \mathbf{x}_2)^2) \\ &= -\mu (1 - (\mathbf{x}_1 \cdot \mathbf{x}_2)^2) - \mu (1 - (\mathbf{x}_2 \cdot \mathbf{x}_1)^2) \\ &= -2\mu \Theta_1 (1 - \Theta_1), \end{aligned} \quad (7.40)$$

and similarly the dynamics of Θ_2 is given by

$$\dot{\Theta}_2 = -2\mu\Theta_2(1 - \Theta_2). \quad (7.41)$$

From (7.40) and (7.41) it becomes obvious that $\dot{\Theta}_i < 0$ whenever $\Theta_i \in (0, 2)$, $i = 1, 2$.

For this flock of 2-agents, wherein each agent employs the TVA strategy, we define the $(2 * 6 - 1)$ -dimensional TVA manifold $\mathcal{M}_{TVA} \subset \mathcal{M}_{state}$ as

$$\begin{aligned} \mathcal{M}_{TVA} &= \{\mathbf{r}_1, [\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1], \mathbf{r}_2, [\mathbf{x}_2, \mathbf{y}_2, \mathbf{z}_2] \in \mathcal{M}_{state} | \Theta_1 \equiv \Theta_2 = 0\} \\ &= \{\mathbf{r}_1, [\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1], \mathbf{r}_2, [\mathbf{x}_2, \mathbf{y}_2, \mathbf{z}_2] \in \mathcal{M}_{state} | \mathbf{x}_1 = \mathbf{x}_2\}. \end{aligned} \quad (7.42)$$

Then, through a way similar to Proposition 7.1, it can be concluded that $\Theta_i(t) \rightarrow 0$ as $t \rightarrow \infty$ whenever $\Theta_i(0) \neq 2$, for $i = 1, 2$. Or in other words, almost any trajectory in \mathcal{M}_{state} converges to \mathcal{M}_{TVA} asymptotically.

7.3 TVA Strategy for an N -agent System with Cyclic Interaction

Now we intend to consider the particular context of N -agent *cyclic interaction* (illustrated in Fig 40) systems in which each agent employs the topological velocity alignment (TVA) strategy. In this setup agent i interacts with agent $(i + 1) \bmod N$, or in other words the neighborhood of agent- i (\mathcal{N}_i) is a singleton set. As a consequence the neighborhood center of mass velocity never ceases to zero (due to regularity of individual trajectories), and each agent aligns its velocity along the velocity of the next agent.

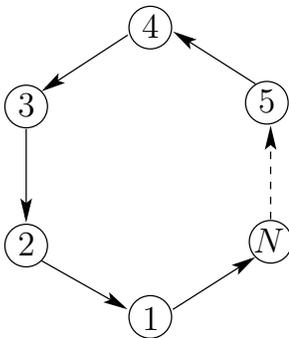


Figure 40: Illustration of an N -agent cyclic interaction system. The direction of influence is conveyed through the direction of arrow-heads.

7.3.1 State Space and Its Reduction onto Shape Space

In an approach similar to the one in Section 7.2.1, we assume our agents to be unit-mass particles tracing out twice continuously-differentiable curves in \mathbb{R}^2 , and model our dynamics using the natural-Frenet frame equations. Therefore our N -agent system can be thought of as evolving on the product space of N copies of the Lie group $SE(2)$. As we are interested in implementing topological velocity alignment (TVA) strategy under a cyclic interaction framework, it is necessary to define the neighbor for agent N . This is done through introduction of an additional element $g_{N+1} \in SE(2)$ to our system state and imposing the closure constraint $g_{N+1} = g_1$. Therefore the state space for our N -agent system can be defined as

$$\mathcal{M}_{state} = \left\{ g_1, g_2, \dots, g_{N+1} \in \underbrace{SE(2) \times SE(2) \times \dots \times SE(2)}_{(N+1)\text{-copies}} \mid \right. \\ \left. g_{N+1} = g_1; g_i e_3 \neq g_{i+1} e_3, i = 1, 2, \dots, N \right\}, \quad (7.43)$$

where $e_3 = [0 \ 0 \ 1]^T$ and $g_i = \begin{bmatrix} \mathbf{x}_i & \mathbf{y}_i & \mathbf{r}_i \\ 0 & 0 & 1 \end{bmatrix}$. In this sense, one can think of our system as a G -snake which *bites its tail* [Krishnaprasad & Tsakiris, 1994].

As we are interested in steering laws (7.6) which leave our system dynamics invariant under rigid motion, we can formulate a reduction to the shape space by defining $\tilde{g}_i \in SE(2)$ as

$$\tilde{g}_i = g_i^{-1} g_{i+1} = \begin{bmatrix} \mathbf{x}_i \cdot \mathbf{x}_{i+1} & \mathbf{x}_i \cdot \mathbf{y}_{i+1} & \mathbf{x}_i \cdot (\mathbf{r}_{i+1} - \mathbf{r}_i) \\ \mathbf{y}_i \cdot \mathbf{x}_{i+1} & \mathbf{y}_i \cdot \mathbf{y}_{i+1} & \mathbf{y}_i \cdot (\mathbf{r}_{i+1} - \mathbf{r}_i) \\ 0 & 0 & 1 \end{bmatrix}, \quad (7.44)$$

and the closure constraint “ $g_{N+1} = g_1$ ” can be expressed in the shape space representation as

$$\prod_{i=1}^N \tilde{g}_i \triangleq \tilde{g}_1 \tilde{g}_2 \dots \tilde{g}_N = \mathbb{1}. \quad (7.45)$$

Therefore the shape space (of relative positions and velocities of the agents) for our N -agent system can be defined as

$$\mathcal{M}_{shape} = \mathcal{M}_{state}/SE(2) = \left\{ \tilde{g}_1, \tilde{g}_2, \dots, \tilde{g}_N \in \underbrace{SE(2) \times SE(2) \times \dots \times SE(2)}_{N\text{-copies}} \mid \prod_{i=1}^N \tilde{g}_i = \mathbb{1}; (\tilde{g}_i)_{13}^2 + (\tilde{g}_i)_{23}^2 \neq 0, i = 1, 2, \dots, N \right\}. \quad (7.46)$$

It has been shown by Justh & Krishnaprasad [2004] that the shape variable \tilde{g}_i

assumes a left-invariant dynamics on $SE(2)$ as

$$\dot{\tilde{g}}_i = \tilde{g}_i \tilde{\xi}_i \tag{7.47}$$

where $\tilde{\xi}_i = \xi_{i+1}(u_{i+1}) - \tilde{g}_i^{-1} \xi_i(u_i) \tilde{g}_i \in \mathfrak{se}(2)$, and the proposed control law (7.6) depends only on the shape variable \tilde{g}_i as

$$u_i = \mu \left[\frac{(\tilde{g}_i)_{21}}{\nu_i} \right]. \tag{7.48}$$

The following result from the work of Galloway [2011] allows us to analyze the shape space dynamics (7.47) as a system of unconstrained dynamics on the product space of N copies of the Lie group $SE(2)$ with the closure constraint (7.45) viewed as a constraint on the initial conditions.

Proposition 7.2 (Proposition 2.2.1, Galloway [2011]). *The constraint $\prod_{i=1}^N \tilde{g}_i = \mathbf{1}$ is preserved by the shape dynamics (7.47).*

7.3.2 Shape Dynamics

Similar to the analysis for a two-agent system (Section 7.2.2), we introduce geometrically meaningful scalar variables by identifying punctured \mathbb{R}^2 with the punctured complex plane. First, we define $\mathbf{r}_{i+1} - \mathbf{r}_i = \rho_i e^{i\theta_i}$ and $\mathbf{x}_i = e^{i\theta_i}$ to parametrize the shape space. Now one can notice that $\tilde{g}_i \in \mathcal{M}_{shape}$ can be represented in terms

of the scalar shape variables as

$$\tilde{g}_i = \begin{bmatrix} \cos \phi_i & \sin \phi_i & -\rho_i \cos \psi_i \\ -\sin \phi_i & \cos \phi_i & \rho_i \sin \psi_i \\ 0 & 0 & 1 \end{bmatrix}, \quad (7.49)$$

where $\phi_i = \theta_i - \theta_{i+1}$ and $\psi_i = \pi - \vartheta_i + \theta_i$ (illustrated in Fig 41).

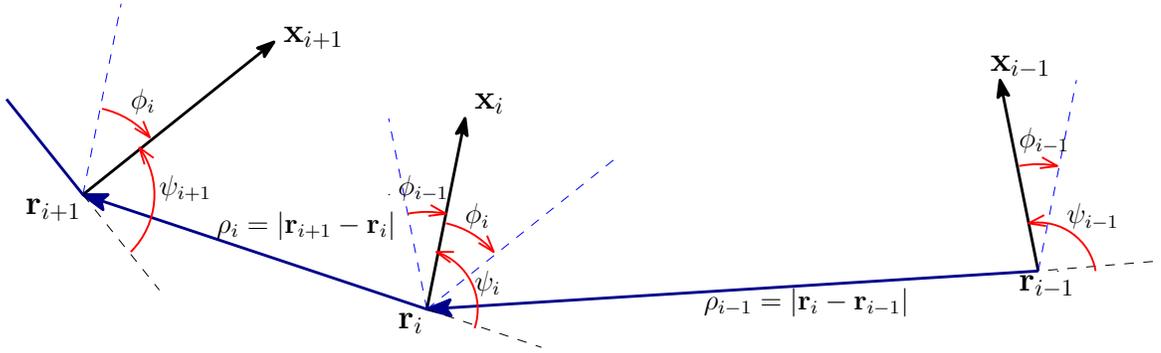


Figure 41: Illustration of scalar shape variables (ρ, ψ, ϕ) used to parametrize the shape space \mathcal{M}_{shape} .

Proposition 7.3 (in a spirit similar to Proposition 2.2.3, Galloway [2011]). *Using shape variables (ρ_i, ψ_i, ϕ_i) the constraint equation (7.45) can be represented as*

$$R\left(\sum_{i=1}^N \phi_i\right) = \mathbf{1} \quad (7.50)$$

$$\sum_{i=1}^N \rho_i R\left(\psi_i + \sum_{j=1}^{i-1} \phi_j\right) = 0, \quad (7.51)$$

where $R(\cdot)$ is a 2×2 rotation matrix.

Proof. The closure constraint (7.45) can alternatively be represented as

$$\tilde{g}_N \tilde{g}_1 \tilde{g}_2 \dots \tilde{g}_{N-1} = \mathbf{1}. \quad (7.52)$$

By defining

$$R(\alpha) = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix}, \quad \alpha \in [0, 2\pi)$$

we can represent $\tilde{g}_i \in \mathcal{M}_{shape}$ as

$$\tilde{g}_i = \begin{bmatrix} R(-\phi_i) & \rho_i R(\pi - \psi_i) e_1 \\ 0 & 1 \end{bmatrix}.$$

Therefore by letting $B_i = R(-\phi_i)$ and $q_i = \rho_i R(\pi - \psi_i) e_1$, we have

$$\begin{aligned} \tilde{g}_N \prod_{i=1}^{N-1} \tilde{g}_i &= \begin{bmatrix} B_N & q_N \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \prod_{i=1}^{N-1} B_i & q_1 + \sum_{i=1}^{N-2} \left(\prod_{j=1}^i B_j \right) q_{i+1} \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} B_N \prod_{i=1}^{N-1} B_i & q_N + B_N q_1 + B_N \sum_{i=1}^{N-2} \left(\prod_{j=1}^i B_j \right) q_{i+1} \\ 0 & 1 \end{bmatrix}. \quad (7.53) \end{aligned}$$

Noting that

$$\begin{aligned}
B_N \sum_{i=1}^{N-2} \left(\prod_{j=1}^i B_j \right) q_{i+1} &= R(-\phi_N) \sum_{i=1}^{N-2} \left(\prod_{j=1}^i R(-\phi_j) \right) \rho_{i+1} R(\pi - \psi_{i+1}) e_1 \\
&= \sum_{i=1}^{N-2} \rho_{i+1} R(-\phi_N) R(\pi - \psi_{i+1} - \sum_{j=1}^i \phi_j) e_1 \\
&= \sum_{i=1}^{N-2} \rho_{i+1} R(\vartheta_{i+1} - \theta_{i+1} - (\theta_1 - \theta_{i+1})) R(-\phi_N) e_1 \\
&= \sum_{i=2}^{N-1} \rho_i R(\vartheta_i - \theta_1) R(-\phi_N) e_1
\end{aligned}$$

and

$$B_N q_1 = R(-\phi_N) \rho_1 R(\pi - \psi_1) e_1 = \rho_1 R(\vartheta_1 - \theta_1) R(-\phi_N) e_1,$$

we have

$$\begin{aligned}
q_N + B_N q_1 + B_N \sum_{i=1}^{N-2} \left(\prod_{j=1}^i B_j \right) q_{i+1} \\
&= \rho_N R(\pi - \psi_N) e_1 + \rho_1 R(\vartheta_1 - \theta_1) R(-\phi_N) e_1 + \sum_{i=2}^{N-1} \rho_i R(\vartheta_i - \theta_1) R(-\phi_N) e_1 \\
&= \rho_N R(\vartheta_N - \theta_1 - \phi_N) e_1 + \sum_{i=1}^{N-1} \rho_i R(\vartheta_i - \theta_1) R(-\phi_N) e_1 \\
&= \sum_{i=1}^N \left[\rho_i R(\pi - \psi_i + \theta_i - \theta_1) \right] R(-\phi_N) e_1 \\
&= - \sum_{i=1}^N \left[\rho_i R^T(\psi_i + \sum_{j=1}^{i-1} \phi_j) \right] R(-\phi_N) e_1. \tag{7.54}
\end{aligned}$$

We also have

$$B_N \prod_{i=1}^{N-1} B_i = R(-\phi_N) \prod_{i=1}^{N-1} R(-\phi_i) = R(-\sum_{i=1}^N \phi_i) = R^T\left(\sum_{i=1}^N \phi_i\right). \tag{7.55}$$

Using (7.54) and (7.55), (7.53) can be simplified into

$$\begin{bmatrix} R^T(\sum_{i=1}^N \phi_i) & -\sum_{i=1}^N \left[\rho_i R^T(\psi_i + \sum_{j=1}^{i-1} \phi_j) \right] R(-\phi_N) e_1 \\ 0 & 1 \end{bmatrix},$$

and the closure constraint (7.45) can alternatively be represented as

$$R\left(\sum_{i=1}^N \phi_i\right) = \mathbf{1} \quad (7.56)$$

$$\sum_{i=1}^N \left[\rho_i R^T(\psi_i + \sum_{j=1}^{i-1} \phi_j) \right] R(-\phi_N) e_1 = 0. \quad (7.57)$$

Now we recall the fact that a linear combination of $SO(2)$ matrices is singular if and only if it is a zero matrix. By using this fact one can conclude from that

$$\sum_{i=1}^N \rho_i R(\psi_i + \sum_{j=1}^{i-1} \phi_j) = 0, \quad (7.58)$$

as $R(-\phi_N) e_1 \neq 0$. □

Now we adopt an approach similar to the one in Section 7.2.2, and derive the dynamics of the scalar shape variables as

$$\begin{aligned} \dot{\rho}_i &= \nu_i \cos \psi_i - \nu_{i+1} \cos(\psi_i - \phi_i) \\ \dot{\psi}_i &= \nu_i u_i - \frac{1}{\rho_i} [\nu_i \sin \psi_i - \nu_{i+1} \sin(\psi_i - \phi_i)] \quad i \in \{1, 2, \dots, N\} \\ \dot{\phi}_i &= \nu_i u_i - \nu_{i+1} u_{i+1}. \end{aligned} \quad (7.59)$$

7.3.3 Analysis of TVA Feedback Law

In this particular context of an N -agent planar, cyclic interaction system, individual contrast functions take the form

$$\begin{aligned}
 \Theta_i &= \frac{1}{2} (\mathbf{x}_{i+1}(t) - \mathbf{x}_i) \cdot (\mathbf{x}_{i+1} - \mathbf{x}_i) \\
 &= 1 - \mathbf{x}_i \cdot \mathbf{x}_{i+1} \\
 &= 1 - \cos \phi_i,
 \end{aligned} \tag{7.60}$$

and the i -th agent is considered to attain the TVA strategy if $\Theta_i = 0$. From (7.60) one can notice that

$$\begin{aligned}
 \Theta_i &\in [0, 2] \\
 \text{and, } \Theta_i = 0 &\Leftrightarrow \phi_i = 0.
 \end{aligned} \tag{7.61}$$

Therefore for this N -agent planar system wherein each agent employs the strategy for topological velocity alignment (TVA), we define the $(2N - 3)$ -dimensional topological velocity alignment manifold $\mathcal{M}_{TVA} \subset \mathcal{M}_{shape}$ as

$$\mathcal{M}_{TVA} = \left\{ \rho_1, \psi_1, \phi_1, \dots, \rho_N, \psi_N, \phi_N \in \mathcal{M}_{shape} \mid \Theta_i = 0, i \in \{1, 2, \dots, N\} \right\}. \tag{7.62}$$

Now we introduce an alternative (holistic) contrast function defined as

$$\Theta_{total} = \sum_{i=0}^N \Theta_i = \sum_{i=0}^N (1 - \cos \phi_i). \tag{7.63}$$

Using the fact given by (7.61), it can be concluded that $\Theta_{total} = 0$ if and only if individual contrast functions are equal to zero. Therefore we can represent the TVA manifold (\mathcal{M}_{TVA}) in terms of the holistic contrast function as

$$\mathcal{M}_{TVA} = \left\{ \rho_1, \psi_1, \phi_1, \dots, \rho_N, \psi_N, \phi_N \in \mathcal{M}_{shape} \mid \Theta_{total} = 0 \right\}. \quad (7.64)$$

Now we focus on a particular steering control law, a planar equivalent of the one given by (7.6), to analyze its effectiveness in attaining the TVA strategy. Moreover from (7.48) we can notice that the feedback law can be expressed in terms of shape variables as

$$u_i = - \left(\frac{\mu}{\nu_i} \right) \sin \phi_i, \quad i \in \{1, 2, \dots, N\}. \quad (7.65)$$

From (7.61) we can observe that once the TVA strategy has been attained, i.e. $\Theta_{total} = 0$, the steering control becomes identically zero, and as a consequence the mismatch in velocity direction remains identically zero (7.59). If each agent employs a steering feedback of the form (7.65), the closed loop dynamics can be expressed as

$$\begin{aligned} \dot{\rho}_i &= \nu_i \cos \psi_i - \nu_{i+1} \cos(\psi_i - \phi_i) \\ \dot{\psi}_i &= -\mu \sin \phi_i - \frac{1}{\rho_i} [\nu_i \sin \psi_i - \nu_{i+1} \sin(\psi_i - \phi_i)] \quad i \in \{1, 2, \dots, N\} \\ \dot{\phi}_i &= \mu (\sin \phi_{i+1} - \sin \phi_i). \end{aligned} \quad (7.66)$$

Proposition 7.4. *The topological velocity alignment manifold $\mathcal{M}_{TVA} \subset \mathcal{M}_{shape}$ is invariant under the closed loop shape dynamics (7.66).*

Proof. From (7.63) and (7.66) we have

$$\begin{aligned}
\dot{\Theta}_{total} &= \mu \sum_{i=0}^N \sin \phi_i (\sin \phi_{i+1} - \sin \phi_i) \\
&= -\frac{\mu}{2} \sum_{i=0}^N (\sin^2 \phi_i - 2 \sin \phi_i \sin \phi_{i+1} + \sin^2 \phi_{i+1}) \\
&= -\frac{\mu}{2} \sum_{i=0}^N (\sin \phi_i - \sin \phi_{i+1})^2.
\end{aligned} \tag{7.67}$$

Therefore it can be concluded that \mathcal{M}_{TVA} is invariant under the shape dynamics because $\dot{\Theta}_{total} = 0 \forall i$ on $\mathcal{M}_{TVA} \subset \mathcal{M}_{shape}$. In fact, from (7.67) it is clear that $\dot{\Theta}_{total} \leq 0$. □

Proposition 7.5. *Consider the set \mathcal{E} defined as*

$$\mathcal{E} = \left\{ (\phi_1, \dots, \phi_N) \in \underbrace{\mathcal{S}^1 \times \dots \times \mathcal{S}^1}_{N\text{-copies}} \mid \dot{\Theta}_{total} = 0, R\left(\sum_{i=1}^N \phi_i\right) = \mathbf{1} \right\}, \tag{7.68}$$

and assume N not to be a multiple of 4. Then \mathcal{E} has a finite cardinality.

Proof. From the closure constraint (7.50) we have

$$\sum_{i=1}^N \phi_i = 2k\pi, \quad k \in \mathbb{Z}. \tag{7.69}$$

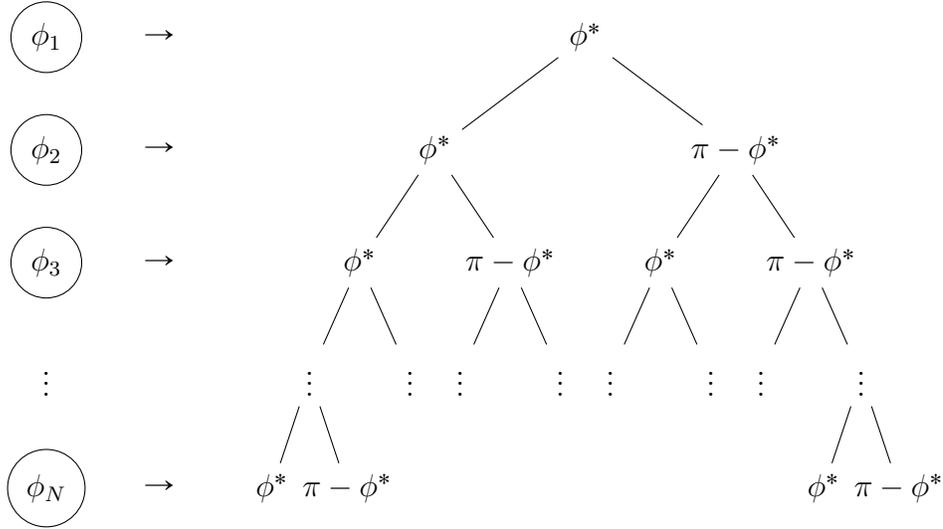
On the other hand, $\dot{\Theta}_{total} = 0$ if and only if

$$\sin \phi_i = \sin \phi_{i+1}, \quad i \in \{1, 2, \dots, N\}, \quad (7.70)$$

and (7.70) holds true if and only if either of the conditions is satisfied

$$\phi_{i+1} = \phi_i, \quad \text{or} \quad \phi_{i+1} = \pi - \phi_i. \quad (7.71)$$

Therefore by assuming $\phi_1 = \phi^*$, we can construct the following tree wherein each branch correspond to a solution for $\dot{\Theta}_{total} = 0$.



Now we assume that along a particular branch of the tree there are M number of ϕ^* -nodes and $(N - M)$ number of $(\pi - \phi^*)$ -nodes. Therefore from (7.69) we have

$$M\phi^* + (N - M)(\pi - \phi^*) = 2k\pi, \quad k \in \mathbb{Z}, \quad \phi^* \in [0, 2\pi), \quad M \in \{1, \dots, N\}. \quad (7.72)$$

Assuming $2M - N \neq 0$ we solve (7.72) to obtain

$$\phi^* = \left(\frac{2k + M - N}{2M - N} \right) \pi, \quad (7.73)$$

and as $\phi^* \in [0, 2\pi)$ we have,

$$k \in \begin{cases} \left[\frac{N - M}{2}, \frac{3M - N}{2} \right) \cap \mathbb{Z} & \text{if, } M > \frac{N}{2} \\ \left[\frac{3M - N}{2}, \frac{N - M}{2} \right) \cap \mathbb{Z} & \text{if, } M < \frac{N}{2}. \end{cases} \quad (7.74)$$

Now if N is a odd number then M can never be equal to $\frac{N}{2}$, and hence for each $M \in \{1, \dots, N\}$, we will have finite number choices for ϕ^* . On the other hand, it can be noted from the tree that a finite number of branches contain M number of ϕ^* -nodes. Combining these two facts we can conclude that \mathcal{E} has finite number of elements.

Now we assume N to be a multiple of 2, but not a multiple of 4. In that case the closure condition (7.69) gets violated for $M = \frac{N}{2}$ because it causes a contradiction in (7.72), and thereby yields no solution for that particular M . For every other value of M , (7.72) yields finite number of ϕ^* . Therefore, the corresponding \mathcal{E} has finite cardinality. \square

However, if N is assumed to be a multiple of 4, then for $M = \frac{N}{2}$, (7.72) holds true over a continuum of ϕ^* . As a consequence \mathcal{E} will have uncountably many elements.

Remark 7.3. *Narrowing our attention to the shape dynamics on \mathcal{M}_{TVA} , we can show that the restricted dynamics can be expressed as*

$$\begin{aligned}\dot{\rho}_i &= (\nu_i - \nu_{i+1}) \cos \psi_i \\ \dot{\psi}_i &= -\frac{1}{\rho_i}(\nu_i - \nu_{i+1}) \sin \psi_i\end{aligned}$$

for every $i \in \{1, 2, \dots, N\}$. This dynamics is essentially same as the restricted dynamics for a two-agent system (7.32), and hence, by following the discussion in Section 7.2.3, we can show that the phase plane trajectories of the reduced dynamics are level sets of “ $\rho_i \sin \psi_i$ ”.

7.4 Algorithm for an N -agent System in a Three-Dimensional Setting

In this section we focus toward topological velocity alignment in its true sense, and assume that each agent pays attention to its K -nearest neighbors. However, in this context of state-dependent attention graph, the possibility of \mathbf{v}_{COM}^i becoming zero cannot be ruled out, and we tackle this issue by bringing in an additional neighbor (for the i -th agent) into consideration whenever \mathbf{v}_{COM}^i becomes zero. As each of the agents has non-zero speed, inclusion of an additional agent into the neighborhood ensures that \mathbf{v}_{COM}^i no longer remains zero. Moreover, it doesn't affect connectivity of the underlying attention graph. The following discrete time algorithm provides a methodical way to implement TVA in a multi-agent robotic system.

It is worth mentioning here that this algorithmic way towards flocking can

Algorithm 2 Topological Velocity Alignment (3D, Nearest Neighbors)

	Initial Time	– $t_{initial}$;
	Final Time	– t_{final} ;
Data:	Sampling Interval	– Δ ;
	Number of Agents	– N ;
	Initial Position and Orientation	– $\{g_i\}_{i=1}^n$;
	Neighborhood Size	– K

begin

Initialize: $t_{current} \leftarrow t_{initial}$

for $i = 1$ *to* n **do**

Initialize: State - $X_i \leftarrow g_i$

while $t_{current} \leq t_{final}$ **do**

for $i = 1$ *to* n **do**

Define: \mathcal{N}_i - the set of K -nearest neighbors

Compute: Neighborhood center of mass velocity - \mathbf{v}_{COM}^i

if $\mathbf{v}_{COM}^i = 0$ **then**

Define: \mathcal{N}_i - the set of $K + 1$ -nearest neighbors

Compute: Neighborhood center of mass velocity - \mathbf{v}_{COM}^i

Compute: Steering control - u_i, v_i

Implement: Steering Control - $\{u_i, v_i\}_{i=1}^n$

Update: State - $\{X_i\}_{i=1}^n$

Update: Time - $t_{current} \leftarrow t_{current} + \Delta$

easily be modified to implement TVA in a planar setting, and this is done by considering the natural curvature u_i alone (v_i is ignored). By restricting (7.6) to a planar setting, u_i can be expressed as

$$u_i(t) = \mu \left[\frac{\mathbf{x}_{\mathcal{N}_i}(t) \cdot \mathbf{y}_i(t)}{\nu_i(t)} \right], \quad (7.75)$$

where all the variables carry their usual meaning.

7.5 Implementation on Mobile Robot Testbed

In what follows we present implementation results of the TVA control law in a planar setting. Our implementation avoids the singularity issue (i.e. $\mathbf{v}_{COM}^i = 0$) by following Algorithm 2.

7.5.1 Experimental Setup

Our experimental test-bed is comprised of Pioneer 3 DX wheeled robots (Fig 42) from Adept MobileRobots [P3-DX]. These compact differential-drive mobile robots are equipped with reversible DC motors, high-resolution motion encoders and 19cm wheels. The onboard computation is done via a 32-bit Renesas SH2-7144 RISC microprocessor, including the P3-SH microcontroller with ARCOS. The sensors on the robot include eight forward-facing ultrasonic (sonar) sensors. ARIA [Aria] provides an interface for controlling and receiving data from the robot, and communication with the robot for sending control commands (*forward velocity* and *turning rate*) is done via 802.11-b/g/n networking. The width of the robot is 380 mm and it has a swing radius of 260 mm.

Algorithm implementation (i.e, feedback law computation) has been done in C++ using ROS along with ROS-ARIA, as the interfacing robotics middleware. The experiments have been carried out in a laboratory environment equipped with a sub-millimeter accurate Vicon motion capture system [ViCoN]. We use a Dell workstation to run ROS, and this computer is connected to the Vicon server via a dedicated Ethernet connection (Fig 43).

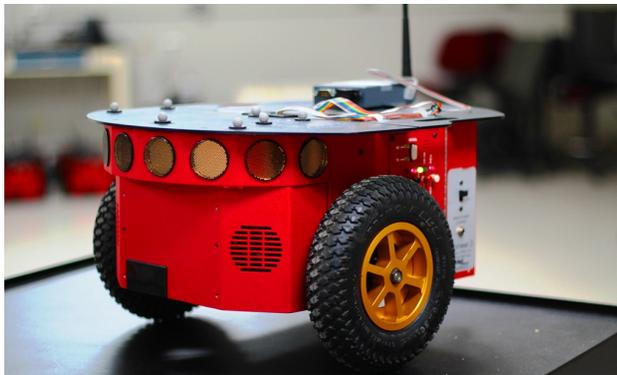


Figure 42: *Pioneer 3-DX Mobile Robot with Two-wheel Differential and Caster.*

The Vicon system captures the motion of the robots and sends out the position and heading data to the computer running ROS. The control law program listens to this data, computes the curvature values, and finally transmits the individual *turning rates* over a Wi-Fi network. All these operations are carried out at a frequency of 25 Hz. As the robot velocity ($\dot{\mathbf{r}}_i^k$, with k denoting the time index) is directed along the robot heading, \mathbf{x}_i^k and \mathbf{y}_i^k can be directly computed from the heading data. Then, the curvature variable u_i^k is evaluated from the corresponding control laws (7.75), and the *turning rate* $\omega_i^k (i = 1, 2, \dots, N)$ (in degrees/sec) is computed as:

$$\omega_i^k = \left(\frac{180}{\pi} \right) \nu_i^k u_i^k, \quad (7.76)$$

where ν_i^k is the speed of the i -th agent at the k -th time instant.

Next we will present our implementation results from three different experiments (refer [YouTube-Video] for implementation videos). In these experiments, the sonar sensors on the robots were activated to sense any obstacle in the direction of motion of the robots and if any robot can sense such an obstacle, it will simply

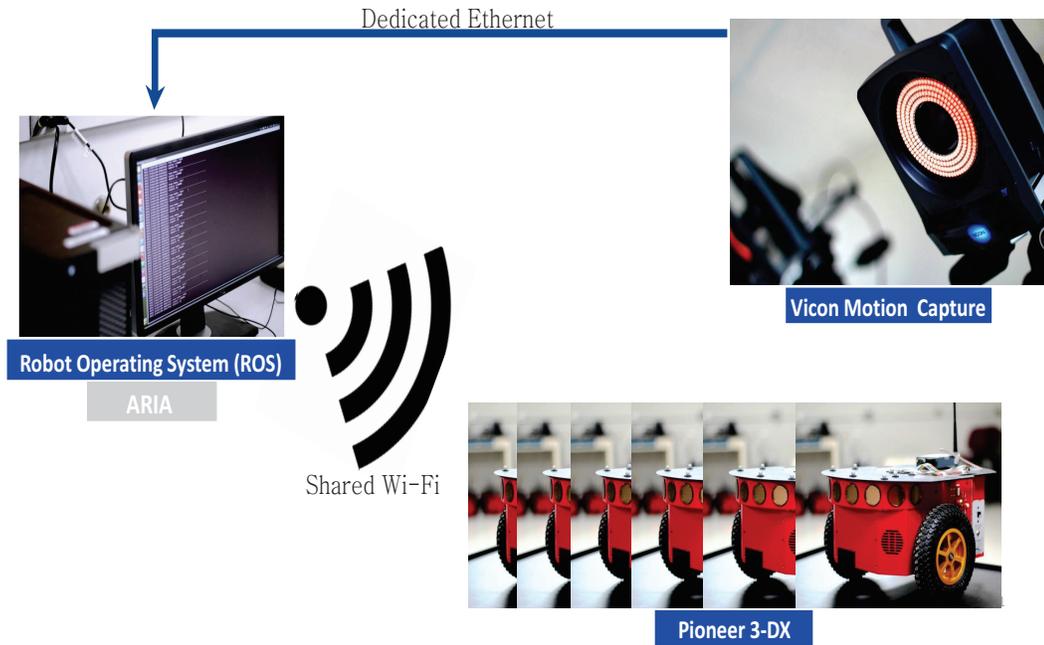


Figure 43: Illustration of the Experimental Setup at Intelligent Servosystems Laboratory, University of Maryland, College Park.

apply a maximum *turning rate* (ω^{sat}) to avoid collision. The sonars are programmed to detect an obstacle only in close proximity (~ 300 mm) of the robots. In all our experiments ω^{sat} is set at be 50 rad/sec, and the value of the feedback gain μ is chosen to be 1 Hz. It should be noted that although the control law allows non-uniform and time-varying forward speed of the robots, here we are presenting sample runs for which the speeds of individual agents are same (60 mm/sec).

7.5.2 Experiment I

A system eight agents is considered, and we apply the same TVA law to all of them. The neighborhood size is taken to be three (i.e. $K = 3$). The robots are initially placed in arbitrary positions and directions. The footprints of the robots

are shown in Fig 44a. The initial and final directions of the robots are shown using arrows and the final positions of the robots are denoted using dots. It can be seen from Fig 44b that the contrast function decays to zero very quickly which indicates perfect velocity alignment within the swarm.

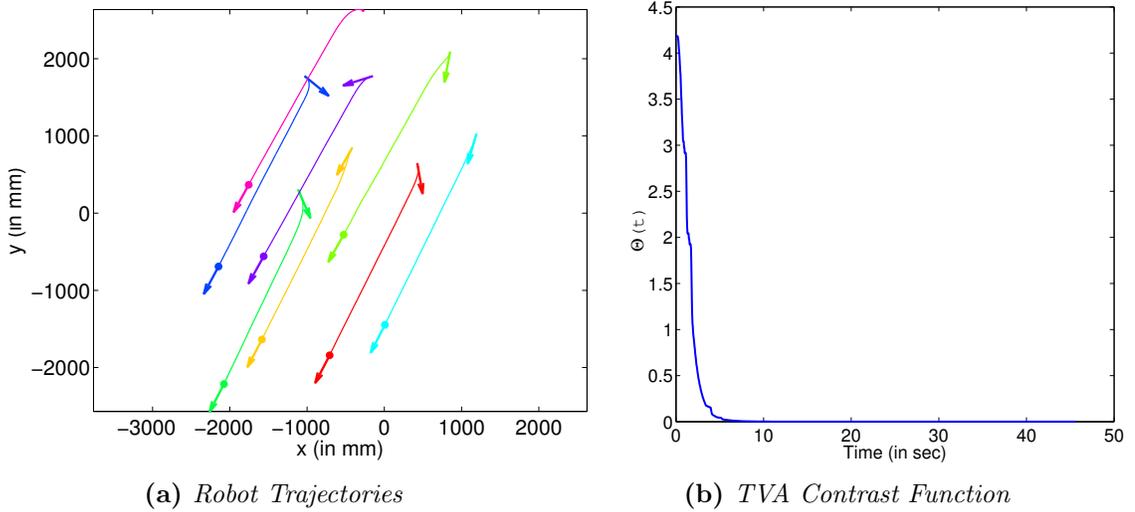


Figure 44: Results from Experiment I (8 agents, Flocking).

7.5.3 Experiment II

Next we decreased the neighborhood size, and set it at $K = 1$, so that each robot ‘communicates’ only with its closest neighbor. We chose the initial positions in such a way that they may form sub-clusters instead of moving as a single swarm. This behavior is called ‘*splitting*’ in a swarm. From Fig 45a, we can clearly see that the swarm of eight robots gradually split from each other and form three different clusters. It is to be noted that even if all the agents are not going in the same direction, the contrast function still converges to zero (Fig 45b). This happens because each of the robots are aligned with their nearest neighbors, and hence each

of the individual contrast functions ($\Theta_i(t)$) are zero. This experiment may explain the splitting phenomenon observable in nature.

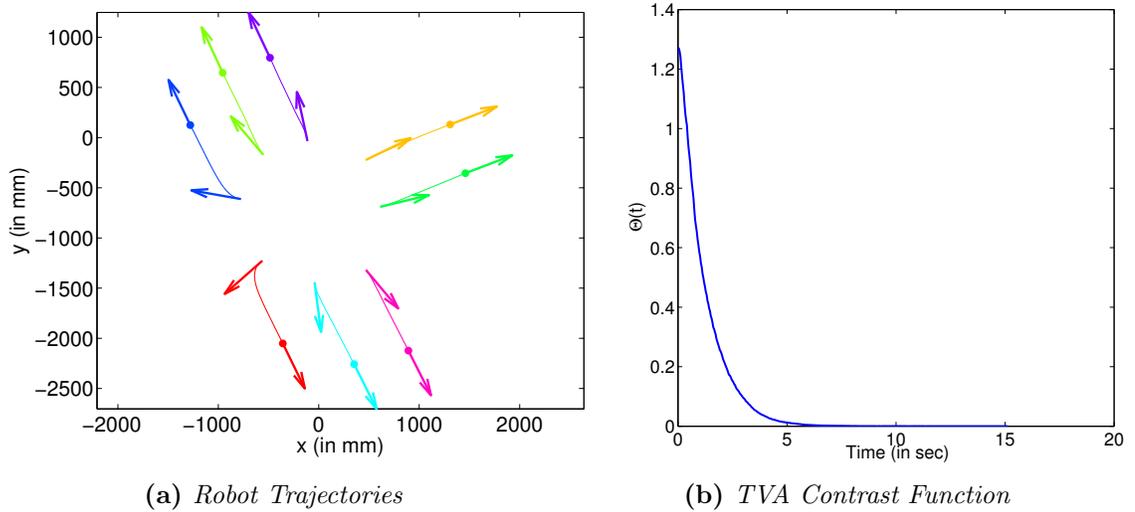


Figure 45: Results from Experiment II (8 agents, Splitting).

7.5.4 Experiment III

Lastly, we combined the above two experiments, and conducted an experiment using six robots in a swarm and another robot as a predator. A separate computer was used for manual control of the ‘predator’ robot.

At the beginning, neighborhood size is set at $K = 3$, such that the ‘communication’ graph among the robots stays connected and they move as an entire swarm in a common direction. When the swarm comes close to the predator, the neighborhood size is decreased to one. As we are not using any onboard visual sensing and the sonar sensing is done only in very close region (~ 300 mm), the change in neighborhood size is made manually. From Fig 46b, we can see that the change in neighborhood size takes place at around 20 seconds and we can also see a tiny jump

in the contrast function value at that time. The predator then slowly approaches to one of the agents in the swarm, which abiding to its collision avoidance rule, turns to avoid the predator. In Fig 46a, the trajectories of the agents are drawn in dashed lines before the occurrence of this event and in solid lines afterwards. The trajectory of the predator robot is not shown in the figure. After creating the initial perturbation, the predator is slowly moved through the swarm causing some subsequent disturbances. These perturbations create a noticeable impact in the swarm. As the attacked agent turns, its neighbor also tries to align itself with that agent and so does its neighbor. This goes on until the communication graph becomes disconnected, and then a split in the swarm is observed (refer [YouTube-Video]) similar to the one in Experiment 2. As we can see in Fig 46a, the swarm is divided in two groups after the attack of the predator. The jumps in the contrast function plot (Fig 46b) symbolize the perturbations caused by the external agent. The contrast function eventually converges to zero after the members are aligned

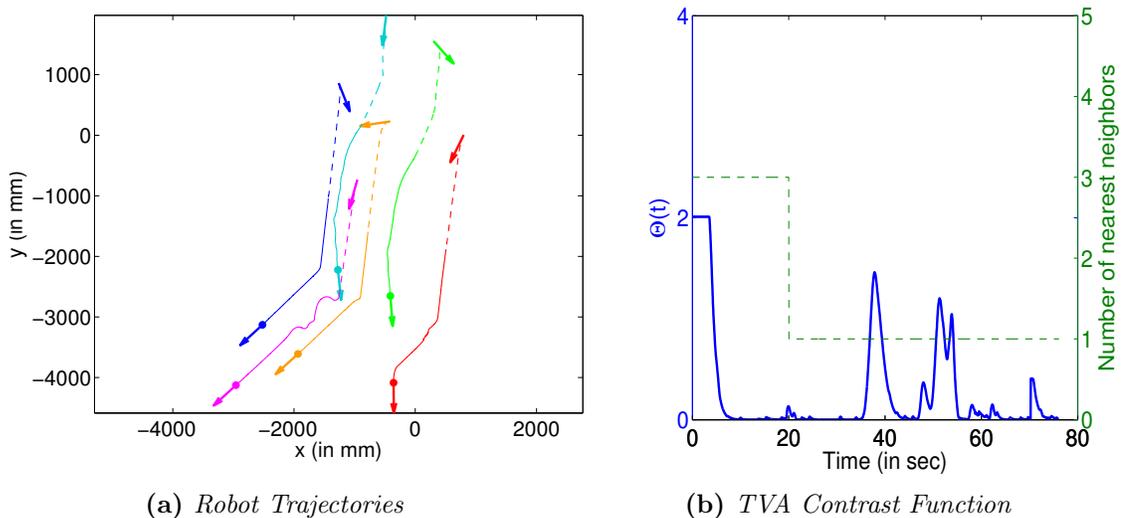


Figure 46: Results from Experiment III (6 agents, Perturbation).

with their neighbors within each subgroup.

Chapter 8: Conclusion and Future Works

The key focus area of this dissertation has been to demonstrate how comprehensive knowledge about the underlying mechanism behind pursuit and collective motion in natural settings can be leveraged to synthesize decentralized control algorithms for collective motion (to be applied to a group of robotic agents). Along the way, we have also developed appropriate algorithms to extract parameters of motion, namely *speed*, *curvatures*, *lateral acceleration etc.*, from a discrete set of observed (perhaps noisy) position data.

In the following section, we summarize the key contributions of this dissertation and propose some topics for further research along these lines.

8.1 Summary of Contributions and Future Directions of Research

8.1.1 Reconstruction of Collectives

As access to the parameters of motion constitutes a necessary step towards uncovering flight strategies and control laws behind collective motion in nature (foraging bats, starling flocks), in “part I” of this dissertation, we identified the need for appropriate algorithms to reconstruct trajectories from a data-set of observed positions. As this problem of recovering a smoothed signal from noisy observations is

an ill-posed one, we introduce regularization to tackle lack of well-posedness. First, to govern the evolution of a trajectory, we introduce generative models (expressed via ODEs) with inputs, states and outputs. Then we impose regularization onto the problem by penalizing appropriate functionals of the control input. Thus we have turned this into a continuous time optimal control problem with intermediate state costs (as shown in Chapter 2). A distinctive feature of this approach lies in the fact that our choice of penalty term has been influenced by findings in bio-mechanics.

In Chapter 2, we have tackled this data smoothing problem from a mathematical programming perspective, and have overcome lack of integrability by adopting a numerical approach. The methodologies developed in this chapter has been used later in Chapter 5 to reconstruct flight trajectories for bat foraging events. We have also proposed an ordinary cross validation approach, based on *leaving-one-out* strategy, to select an optimal amount of regularization, which plays a crucial role in maintaining a proper balance between goodness of fit and smoothness of the trajectory.

However, as leaving-one-out requires the reconstruction to be carried out multiple times (same as the data-set size) by dropping out a single observation every time, implementation of ordinary cross validation is quite demanding from a computational perspective. It would be interesting to investigate if a computationally efficient alternative can replace this algorithmic approach for optimal amount of regularization. Unbiased risk estimators (Li [1985]; Solo [1996]) based on Charles Stein's work [Stein, 1981] on mean estimation for multivariate normal distribution, appears very relevant in our context.

In Chapter 3, we have introduced a linear generative model (basically a triple integrator), and imposed regularization by trading total fit error against high values of jerk (*i.e.* third derivative of position) path integral. Then, by exploiting integrability of the generative model and quadratic nature of the cost functional, we have derived a closed form solution. Moreover, we have shown that the reconstructed position, velocity and acceleration can be expressed a linear combination of the observed position data. By choosing appropriate linear generative model, one can easily show that smoothing splines can be posed as a special case in this framework. The trajectory reconstruction algorithm developed in this chapter has later been used in Chapter 6 to reconstruct flight trajectories of starling flocks.

Our numerical example in this chapter (Section 3.5) attempted to reconstruct a curve on a sphere from a set of discrete and noisy observations. Although this approach yields a satisfactory performance in terms of enforcing the reconstructed trajectory to lie on the sphere, it would be exciting to construct data smoothing algorithms to reconstruct trajectories which lie on a lower dimensional algebraic manifold in the ambient space. Another potential extension of this work lies in the area of reconstructing periodic curves.

In Chapter 4, we have exploited the theory of Pontryagin's maximum principle to solve data smoothing posed as a continuous time optimal control problem. The proposed results are capable of dealing with data smoothing problems in both Euclidean (\mathbb{R}^n) and matrix Lie group (G) settings, and they yield result in a semi-analytic way. Example problem demonstrates that this theory enables us to turn an optimal control problem over an infinite dimensional function space into a two-point

boundary value problem, which can be tackled via an appropriate multiple-shooting method [Morrison *et al.*, 1962]. It is worth mentioning here that this regularized inversion problem can also be viewed from the perspective of waypoint tracking. If someone attempts to achieve reduction in some path cost, by sacrificing exactness in its traversal of way-points, that problem can easily be cast in this framework.

However, there are some issues which requires attention before we attempt to broaden the scope of this framework. One obvious direction along this line is to consider penalty functionals involving derivatives of control inputs. Although this issue can be addressed by augmenting new states to the system, some care should be taken to set up the associated symplectic framework in a proper way. Another pressing concern lies in the singularity issues of the Hamiltonian. The following data smoothing problem exemplifies this issue in an efficient way.

Let us consider the generative model, governed by the natural Frenet frame equations, for evolution of a trajectory in a three-dimensional space. As discussed in Section 1.3.1, we can pack the position vector and the associated frame vectors, inside a 4×4 matrix $g(t)$ defined as

$$g(t) = \begin{bmatrix} \mathbf{x}(t) & \mathbf{y}(t) & \mathbf{z}(t) & \mathbf{r}(t) \\ 0 & 0 & 0 & 1 \end{bmatrix} \in SE(3).$$

Now, by letting

$$\begin{aligned}
 X_1 &= \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, & X_2 &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, & X_3 &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \\
 X_4 &= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, & X_5 &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, & X_6 &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix},
 \end{aligned}$$

denote a basis for the associated Lie algebra $\mathfrak{se}(3)$, the underlying generative model can be expressed as the following left-invariant dynamical system on $SE(3)$,

$$\dot{g} = g\xi_u = g(\nu(X_4 + uX_1 - vX_2)) \quad (8.1)$$

where u, v are natural curvatures (steering control) and ν is the speed of the trajectory. Then, by letting $\{r_i\}_{i=0}^N$ denote the set of observed positions, and by imposing regularization via trading total fit error against high values of the curvatures and speed path integral, one can formulate the trajectory reconstruction problem as the

following optimal control problem

$$\begin{aligned} \text{Minimize}_{g(t_0), u, v, \nu} \quad & \sum_{i=0}^N \|\mathbf{r}(t_i) - r_i\|^2 + \lambda \int_{t_0}^{t_N} (u^2 + v^2 + \nu^2) dt \\ \text{subject to} \quad & g(t_0) \in SE(3), \quad u, v \in \mathcal{U}, \quad \nu \in \mathcal{U}^+ \end{aligned} \tag{8.2}$$

$$\dot{g} = g(\nu(X_4 + uX_1 - vX_2)) = T_e L_g \cdot (\nu(X_4 + uX_1 - vX_2)),$$

where $\mathcal{U}(\mathcal{U}^+)$ is the space of real(positive) valued functions on $[t_0, t_N]$ and $\lambda > 0$ acts as a regularization parameter for the inverse problem. By comparing this optimal control problem (8.2) with the one mentioned in the statement of maximum principle (4.45) we have

$$\begin{aligned} L(u) &= \lambda(u^2 + v^2 + \nu^2) \\ f(g(t_i), r_i) &= \|Ag(t_i)e_4 - r_i\|^2 \end{aligned}$$

where $A = [e_1 \ e_2 \ e_3]^T$ and $\{e_i\}_{i=1}^4$ denotes a standard basis vector in \mathbb{R}^4 . By following an approach similar to the one adopted for the previous example problems, we define the pre-Hamiltonian as

$$H(g, p, u) = \langle p, T_e L_g \cdot \xi_u \rangle - L(u) \tag{8.3}$$

where $p \in T_g^* SE(3)$ and $T_e L_g$ represents the tangent map of the left translation by g on $SE(3)$. Now we introduce $\mu \in \mathfrak{se}^*(3)$ defined as $\mu = T_e L_g^* \cdot p$. By letting

X_i^b , $i = 1, \dots, 6$ denote a dual basis for $\mathfrak{se}^*(3)$, μ can be represented as

$$\mu = \sum_{i=1}^6 \mu_i X_i^b.$$

Then, by leveraging the left-invariance of the dynamics (8.1), the pre-hamiltonian can be expressed as

$$\begin{aligned} H(g, p, u) &= \langle T_e L_g^* \cdot p, \xi_u \rangle - L(u) \\ &= \left\langle \sum_{i=1}^6 \mu_i X_i^b, \nu(X_4 + uX_1 - vX_2) \right\rangle - L(u) \\ &= \nu(\mu_4 + u\mu_1 - v\mu_2) - \lambda(u^2 + v^2 + \nu^2). \end{aligned} \quad (8.4)$$

As both ξ_u and $L(u)$ are differentiable with respect to the control inputs, the optimal control can be derived by solving

$$\begin{aligned} \left. \frac{\partial H}{\partial u} \right|_{(u^*, v^*, \nu^*)} &= \nu^* \mu_1 - 2\lambda u^* = 0 \\ \left. \frac{\partial H}{\partial v} \right|_{(u^*, v^*, \nu^*)} &= -\nu^* \mu_2 - 2\lambda v^* = 0 \\ \left. \frac{\partial H}{\partial \nu} \right|_{(u^*, v^*, \nu^*)} &= \mu_4 + u^* \mu_1 - v^* \mu_2 - 2\lambda \nu^* = 0. \end{aligned} \quad (8.5)$$

Hence the optimal control inputs can be expressed by

$$\begin{pmatrix} u^* \\ v^* \\ \nu^* \end{pmatrix} = \frac{1}{4\lambda^2 - (\mu_1^2 + \mu_2^2)} \begin{pmatrix} \mu_1 \mu_4 \\ -\mu_2 \mu_4 \\ 2\lambda \mu_4 \end{pmatrix}, \quad (8.6)$$

and by substituting the optimal controls into the pre-hamiltonian, (4.87) yields an $SE(2)$ -invariant reduced hamiltonian, given by

$$h = \frac{\lambda\mu_4^2}{4\lambda^2 - (\mu_1^2 + \mu_2^2)}. \quad (8.7)$$

Now, we can derive the reduced dynamics on $\mathfrak{se}^*(3)$ by computing $\nabla_\mu h$ and the associated structure constants. However from (8.7) one can notice that the reduced hamiltonian becomes singular whenever $\mu_1^2 + \mu_2^2 = 4\lambda^2$. With this perspective, it would be interesting to seek answer for such questions as:

- Is the submanifold $\mathcal{M}_{sing} = \{(\mu_1, \dots, \mu_6) \in \mathbb{R}^6 | \mu_1^2 + \mu_2^2 = 4\lambda^2\}$ invariant under the reduced dynamics?
- Does this submanifold (\mathcal{M}_{sing}) attract any trajectory originating outside the submanifold? Under what conditions one can avoid the singularity?

8.1.2 Analysis of Collective Behavior in Nature

Following the main flow of this dissertation, we set out to explore the underlying mechanism behind pursuit and collective motion in natural settings in “part II” of this dissertation, and performed analysis on the flight trajectory data of echolocating bats (demonstrating pursuit) and European starlings (demonstrating flocking). Our study adopts a two-pronged approach to investigate the underlying flight strategies and feedback mechanism for steering control - first we study the statistics of appropriate contrast functions, and then we compare the empirical values of steer-

ing control to the values obtained from theoretically plausible feedback laws. This approach is also capable of providing an estimate for the associated sensorimotor delay.

In Chapter 5, we have performed flight data analysis for big brown bats (*Eptesicus fuscus*) in two different foraging contexts. This analysis has shown quantitative evidence in favor of a context-specific switch in flight strategy. According to our findings, bats apply constant absolute target direction (CATD), also known as motion camouflage (MC) in the context of visual insects, while chasing a free flying insect (praying mantis in our study). But if the scenario is modified into a competitive setting with another bat foraging for the same stationary food source (meal-worm in this set of experiments), flight data show evidence that the trailing bat resort to classical pursuit (CP) to follow the other bat. Moreover, by comparing empirically observed curvature values with the ones obtained from theoretically plausible feedback laws, this study sheds light upon the steering control mechanisms and the associated sensorimotor delay.

Recent developments by Galloway & Dey [2015] in decentralized control have analyzed cyclic constant bearing (CB) pursuit [Galloway *et al.*, 2013] strategy in a multi-agent system wherein each agent pays attention to a neighbor (moving) and a beacon (fixed). As classical pursuit can be interpreted as a special case of the CB pursuit strategy, it would be worthwhile to study this interaction between the stationary prey (tethered meal-worm) and the flying bats from this perspective (each bat having its own priority level for the stationary food source). It is worth mentioning here that our analysis does not assume any internal model of target

motion, and the steering control mechanism is assumed to be based on pure reaction to the target motion. However a recent study by Mischiati *et al.* [2014] has shown that, internal model and reactive pursuit strategy, both play an important role in generating the prey interception trajectories by dragonflies (*Plathemis lydia*). So it would be interesting to design appropriate experiments and investigate if such internal models are involved in bat foraging too.

Chapter 6 describes our ongoing work on the flight strategy analysis of European starling (*Sturnus vulgaris*) flocks. This study has revealed that the flock-averaged coherence (the average cosine of the angle between the velocities of a focal bird and its neighborhood center of mass, averaged over the entire flock) gets maximized by considering 5-7 nearest neighbors. In addition to reconfirming a previous result highlighting the importance of topological notion of distance in starling flocks, this study has also provided some insight about the steering control actions adopted by the individual starlings.

However, our current approach for constructing an interaction graph (based on a set of nearest neighbors) is oblivious to sensory perceptions by the focal individual and spatial distribution of the neighbors. Future works will attempt to explore other possibilities for the interaction graph based on visual cues [Strandburg-Peshkin *et al.*, 2013] and statistical causality (Granger causality [Granger, 1969], directed mutual information [Massey, 1990]).

8.1.3 Synthesis of Collective Motion

Statistical analysis of flight behavior in European starling flocks has revealed that the individuals in a flock tend to fly parallel to each other. This observation led us towards proposing a decentralized algorithm (called topological velocity alignment) to make the individuals move in the same direction without colliding into each other (in Chapter 7), and the global behavior emerges through local interaction between neighbors. The fact, that the proposed feedback law does not assume any uniformity in the individual speeds and the attention graph is directed, makes our approach distinct from the existing models of flocking. Moreover, this proposed control strategy has wave-like aspects, conforming to the criterion of linear growth of information observed in starling flocks [Attanasi *et al.*, 2014].

Numerical simulations and implementations have shown that whenever a subgroup of agents in the flock are provided additional instruction (e.g. to follow predefined paths, avoid obstacles, etc.) for steering control, the effect spreads across the whole flock (without any change in the way the uninformed agents interact with its neighbors). These observations demand further analysis, and future work in this area will explore how these covert leaders (agents with extra information) in a collaborative swarm affect the group behavior. This framework can also be exploited to design a collaboration of heterogeneous agents wherein a small group of agents are capable of sensing the environmental cue and others just interact with their neighbors.

Bibliography

- Ahmadzadeh, A., Jadbabaie, A., Kumar, V. & Pappas, G. J. 2006 Multi-UAV Cooperative Surveillance with Spatio-Temporal Specifications. In *Proceedings of the 45th IEEE Conference on Decision and Control (CDC)*, pp. 5293–5298. San Diego, CA. (doi:10.1109/CDC.2006.377157)
- Allen, D. M. 1974 The Relationship between Variable Selection and Data Augmentation and a Method for Prediction. *Technometrics*, **16**(1), 125–127. (doi:10.2307/1267500)
- Andriluka, M., Schnitzspan, P., Meyer, J., Kohlbrecher, S., Petersen, K., von Stryk, O., Roth, S. & Schiele, B. 2010 Vision Based Victim Detection from Unmanned Aerial Vehicles. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1740–1747. Taipei, Taiwan. (doi:10.1109/IROS.2010.5649223)
- Aoki, I. 1982 A Simulation Study on the Schooling Mechanism in Fish. *Bulletin of the Japanese Society of Scientific Fisheries*, **48**(8), 1081–1088. (doi:10.2331/suisan.48.1081)
- Aria ROS-ARIA Package. <http://wiki.ros.org/ROSARIA>. Accessed: 2014-09-07.

- Attanasi, A., Cavagna, A., Castello, L. D., Giardina, I., Grigera, T. S., Jelic, A., Melillo, S., Parisi, L., Pohl, O. *et al.* 2014 Information Transfer and Behavioural Inertia in Starling Flocks. *Nature Physics*, **10**(9), 691–696. (doi:10.1038/nphys3035)
- Attanasi, A. and Cavagna, A. and Del Castello, L. and Giardina, I. and Jelic, A. and Melillo, S. and Parisi, L. and Shen, E. and Silvestri, E. and Viale, M. 2013 Tracking in Three Dimensions via Multi-path Branching. *ArXiv e-prints*.
- Ballerini, M., Cabibbo, N., Candelier, R., Cavagna, A., Cisbani, E., Giardina, I., Lecomte, V., Orlandi, A., Parisi, G. *et al.* 2008a Interaction Ruling Animal Collective Behavior Depends on Topological Rather Than Metric Distance: Evidence from a Field Study. *Proceedings of the National Academy of Sciences*, **105**(4), 1232–1237. (doi:10.1073/pnas.0711437105)
- Ballerini, M., Cabibbo, N., Candelier, R., Cavagna, A., Cisbani, E., Giardina, I., Orlandi, A., Parisi, G., Procaccini, A. *et al.* 2008b Empirical Investigation of Starling Flocks: A Benchmark Study in Collective Animal Behaviour. *Animal Behaviour*, **76**(1), 201–215. (doi:10.1016/j.anbehav.2008.02.004)
- Bazazi, S., Buhl, J., Hale, J. J., Anstey, M. L., Sword, G. A., Simpson, S. J. & Couzin, I. D. 2008 Collective Motion and Cannibalism in Locust Migratory Bands. *Current Biology*, **18**(10), 735–739. (doi:10.1016/j.cub.2008.04.035)
- Beekman, M., Sumpter, D. J. T. & Ratnieks, F. L. W. 2001 Phase Transition between Disordered and Ordered Foraging in Pharaoh’s Ants. *Proceedings of the National Academy of Sciences*, **98**(17), 9703–9706. (doi:10.1073/pnas.161285298)
- Ben-Yosef, G. & Ben-Shahar, O. 2012 A Tangent Bundle Theory for Visual Curve Completion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **34**(7), 1263–1280. (doi:10.1109/TPAMI.2011.262)

- Bertsekas, D. P. & Tsitsiklis, J. N. 2007 Comments on “Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rules”. *IEEE Transactions on Automatic Control*, **52**(5), 968–969. (doi:10.1109/TAC.2007.895885)
- Bethke, B., How, J. P. & Vian, J. 2009 Multi-UAV Persistent Surveillance With Communication Constraints and Health Management. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference (GNC)*. Chicago, IL. (doi:10.2514/6.2009-5654)
- Bhagavatula, P. S., Claudianos, C., Ibbotson, M. R. & Srinivasan, M. V. 2011 Optic Flow Cues Guide Flight in Birds. *Current Biology*, **21**(21), 1794–1799. (doi:10.1016/j.cub.2011.09.009)
- Bishop, R. L. 1975 There is More than One Way to Frame a Curve. *The American Mathematical Monthly*, **82**(3), 246–251. (doi:10.2307/2319846)
- Brockett, R. 1970 *Finite Dimensional Linear Systems*. New York, NY: Wiley.
- Brody, D. C., Holm, D. D. & Meier, D. M. 2012 Quantum Splines. *Physical Review Letters*, **109**, 100 501. (doi:10.1103/PhysRevLett.109.100501)
- Burnett, C. L., Holm, D. D. & Meier, D. M. 2013 Inexact Trajectory Planning and Inverse Problems in the Hamilton-Pontryagin Framework. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **469**(2160), 20130 249. (doi:10.1098/rspa.2013.0249)
- Carere, C., Montanino, S., Moreschini, F., Zoratto, F., Chiarotti, F., Santucci, D. & Alleva, E. 2009 Aerial Flocking Patterns of Wintering Starlings, *Sturnus Vulgaris*, under Different Predation Risk. *Animal Behaviour*, **77**(1), 101 – 107. (doi:http://dx.doi.org/10.1016/j.anbehav.2008.08.034)

- Cavagna, A., Cimorelli, A., Giardina, I., Parisi, G., Santagati, R., Stefanini, F. & Viale, M. 2010 Scale-free Correlations in Starling Flocks. *Proceedings of the National Academy of Sciences*, **107**(26), 11 865–11 870. (doi:10.1073/pnas.1005766107)
- Cavagna, A., Giardina, I., Orlandi, A., Parisi, G. & Procaccini, A. 2008a The STARFLAG Handbook on Collective Animal Behaviour: 2. Three-dimensional Analysis. *Animal Behaviour*, **76**(1), 237–248. (doi:10.1016/j.anbehav.2008.02.003)
- Cavagna, A., Giardina, I., Orlandi, A., Parisi, G., Procaccini, A., Viale, M. & Zdrvkovic, V. 2008b The STARFLAG Handbook on Collective Animal Behaviour: 1. Empirical Methods. *Animal Behaviour*, **76**(1), 217–236. (doi:10.1016/j.anbehav.2008.02.002)
- Chiu, C., Reddy, P. V., Xian, W., Krishnaprasad, P. S. & Moss, C. F. 2010 Effects of Competitive Prey Capture on Flight Behavior and Sonar Beam Pattern in Paired Big Brown Bats, *Eptesicus Fuscus*. *Journal of Experimental Biology*, **213**(19), 3348–3356. (doi:10.1242/jeb.044818)
- Chiu, C., Xian, W. & Moss, C. F. 2008 Flying in Silence: Echolocating Bats Cease Vocalizing to Avoid Sonar Jamming. *Proceedings of the National Academy of Sciences*, **105**(35), 13 116–13 121. (doi:10.1073/pnas.0804408105)
- Crouch, P. & Leite, F. S. 1991 Geometry and the Dynamic Interpolation Problem. In *Proceedings of the American Control Conference (ACC)*, pp. 1131–1136. Boston, MA.
- Crouch, P. & Leite, F. S. 1995 The Dynamic Interpolation Problem: On Riemannian Manifolds, Lie Groups, and Symmetric Spaces. *Journal of Dynamical and Control Systems*, **1**(2), 177–202. (doi:10.1007/BF02254638)

- Cucker, F. & Smale, S. 2007 Emergent Behavior in Flocks. *IEEE Transactions on Automatic Control*, **52**(5), 852 –862. (doi:10.1109/TAC.2007.895842)
- Cutts, C. & Speakman, J. 1994 Energy Savings in Formation Flight of Pink-footed Geese. *Journal of Experimental Biology*, **189**(1), 251 – 261.
- Dey, B. & Krishnaprasad, P. S. 2012 Trajectory Smoothing as a Linear Optimal Control Problem. In *Proceedings of the 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1490–1497. Monticello, IL. (doi:10.1109/Allerton.2012.6483395)
- Dey, B. & Krishnaprasad, P. S. 2014a Control-Theoretic Data Smoothing. In *Proceedings of 53rd IEEE Conference on Decision and Control (CDC)*, pp. 5064–5070. Los Angeles, CA.
- Dey, B. & Krishnaprasad, P. S. 2014b Trajectory Reconstruction via Optimal Control. To be submitted.
- Dey, B., Reddy, P. V., Chiu, C., Ghose, K., Galloway, K. S., Horiuchi, T. K., Justh, E. W., Moss, C. F. & Krishnaprasad, P. S. 2014 Context Shapes Bat Flight Strategy: evidence from the geometry of prey capture. To be submitted.
- do Carmo, M. P. 1976 *Differential Geometry of Curves and Surfaces*. Englewood Cliffs, NJ: Prentice-Hall Inc.
- Dunbabin, M. & Marques, L. 2012 Robots for Environmental Monitoring: Significant Advancements and Applications. *IEEE Robotics Automation Magazine*, **19**(1), 24–39. (doi:10.1109/MRA.2011.2181683)
- Egerstedt, M. & Martin, C. 2010 *Control Theoretic Splines: Optimal Control, Statistics, and Path Planning*. Princeton, NJ: Princeton University Press.

- Elfes, A., Bueno, S. S., Bergerman, M. & Ramos, J. G. 1998 A Semi-Autonomous Robotic Airship for Environmental Monitoring Missions. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 4, pp. 3449–3455. Leuven, Belgium. (doi:10.1109/ROBOT.1998.680971)
- Flash, T. & Hogan, N. 1985 The Coordination of Arm Movements: An Experimentally Confirmed Mathematical Model. *The Journal of Neuroscience*, **5**(7), 1688–1703.
- fminunc MATLAB Documentation. <http://ter.ps/fminuncdoc>. Accessed: 2012-07-23.
- Galloway, K. S. 2011 *Cyclic Pursuit: Symmetry, Reduction and Nonlinear Dynamics*. PhD Thesis: University of Maryland, College Park.
- Galloway, K. S. & Dey, B. 2015 Station Keeping through Beacon-referenced Cyclic Pursuit. Accepted for American Control Conference (ACC 2015).
- Galloway, K. S., Justh, E. & Krishnaprasad, P. S. 2010 Cyclic Pursuit in Three Dimensions. In *Proceedings of the 49th IEEE Conference on Decision and Control (CDC)*, pp. 7141–7146. Atlanta, GA. (doi:10.1109/CDC.2010.5717459)
- Galloway, K. S., Justh, E. W. & Krishnaprasad, P. S. 2013 Symmetry and Reduction in Collectives: Cyclic Pursuit Strategies. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **469**(2158), 20130 264. (doi:10.1098/rspa.2013.0264)
- Gautrais, J., Jost, C., Soria, M., Campo, A., Motsch, S., Fournier, R., Blanco, S. & Theraulaz, G. 2009 Analyzing Fish Movement as a Persistent Turning Walker. *Journal of Mathematical Biology*, **58**(3), 429–445. (doi:10.1007/s00285-008-0198-7)

- Ghose, K., Horiuchi, T. K., Krishnaprasad, P. S. & Moss, C. F. 2006 Echolocating Bats Use a Nearly Time-Optimal Strategy to Intercept Prey. *PLoS Biology*, **4**(5), 865–873. (doi:10.1371/journal.pbio.0040108)
- Ghose, K. & Moss, C. F. 2006 Steering by Hearing: A Bat's Acoustic Gaze Is Linked to its Flight Motor Output by a Delayed, Adaptive Linear Law. *The Journal of Neuroscience*, **26**(6), 1704–1710. (doi:10.1523/JNEUROSCI.4315-05.2006)
- Granger, C. W. J. 1969 Investigating Causal Relations by Econometric Models and Cross-spectral Methods. *Econometrica*, **37**(3), 890–893. (doi:10.2307/1912791)
- Griffin, D. R., Webster, F. A. & Michael, C. R. 1960 The Echolocation of Flying Insects by Bats. *Animal Behaviour*, **8**(3-4), 141–154. (doi:10.1016/0003-3472(60)90022-1)
- Halder, U. & Dey, B. 2015 Biomimetic Algorithms for Coordinated Motion: Theory and Implementation. Accepted for International Conference on Robotics and Automation (ICRA 2015).
- Harmon, S. 1987 The Ground Surveillance Robot (GSR): An Autonomous Vehicle Designed to Transit Unknown Terrain. *IEEE Journal of Robotics and Automation*, **3**(3), 266–279. (doi:10.1109/JRA.1987.1087091)
- Herskin, J. & Steffensen, J. F. 1998 Energy Savings in Sea Bass Swimming in a School: Measurements of Tail Beat Frequency and Oxygen Consumption at Different Swimming Speeds. *Journal of Fish Biology*, **53**(2), 366 – 376. (doi: <http://dx.doi.org/10.1111/j.1095-8649.1998.tb00986.x>)
- Hubel, D. H. 1995 *Eye, Brain, and Vision*. No. 22 in Scientific American Library Series. New York, NY: W. H. Freeman.

- Jadbabaie, A., Lin, J. & Morse, S. A. 2003 Coordination of Groups of Mobile Autonomous Agents using Nearest Neighbor Rules. *IEEE Transactions on Automatic Control*, **48**(6), 988 – 1001. (doi:10.1109/TAC.2003.812781)
- Jakubiak, J., Leite, F. S. & Rodrigues, R. C. 2006 A Two-step Algorithm of Smooth Spline Generation on Riemannian Manifolds. *Journal of Computational and Applied Mathematics*, **194**(2), 177 – 191. (doi:http://dx.doi.org/10.1016/j.cam.2005.07.003)
- Justh, E. W. & Krishnaprasad, P. S. 2002 A Simple Control Law for UAV Formation Flying. Tech. Rep. TR 2002-38, Institute for Systems Research, University of Maryland, College Park, MD.
- Justh, E. W. & Krishnaprasad, P. S. 2003 Steering Laws and Continuum Models for Planar Formations. In *Proceedings of the 42nd IEEE Conference on Decision and Control (CDC)*, pp. 3609–3615. Maui, HI. (doi:10.1109/CDC.2003.1271708)
- Justh, E. W. & Krishnaprasad, P. S. 2004 Equilibria and Steering Laws for Planar Formations. *Systems & Control Letters*, **52**(1), 25–38. (doi:http://dx.doi.org/10.1016/j.sysconle.2003.10.004)
- Justh, E. W. & Krishnaprasad, P. S. 2006 Steering Laws for Motion Camouflage. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **462**(2076), 3629–3643. (doi:10.1098/rspa.2006.1742)
- Justh, E. W. & Krishnaprasad, P. S. 2011 Optimal Natural Frames. *Communications in Information and Systems*, **11**(1), 17–34. (doi:10.4310/CIS.2011.v11.n1.a2)
- Kane, S. A. & Zamani, M. 2014 Falcons Pursue Prey Using Visual Motion Cues: New Perspectives from Animal-borne Cameras. *Journal of Experimental Biology*, **217**(2), 225–234. (doi:10.1242/jeb.092403)

- Kano, H., Egerstedt, M., Fujioka, H., Takahashi, S. & Martin, C. 2008 Periodic smoothing splines. *Automatica*, **44**(1), 185 – 192. (doi:10.1016/j.automatica.2007.05.011)
- Kingston, D., Beard, R. W. & Holt, R. S. 2008 Decentralized Perimeter Surveillance Using a Team of UAVs. *IEEE Transactions on Robotics*, **24**(6), 1394–1404. (doi:10.1109/TRO.2008.2007935)
- Krishnaprasad, P. S. 1985 Lie-Poisson Structures, Dual-Spin Spacecraft and Asymptotic Stability. *Nonlinear Analysis: Theory, Methods and Applications*, **9**(10), 1011–1035. (doi:10.1016/0362-546X(85)90083-5)
- Krishnaprasad, P. S. 1993 Optimal Control and Poisson Reduction. Tech. Rep. TR 93-87, Institute for Systems Research, University of Maryland, College Park, MD.
- Krishnaprasad, P. S. & Tsakiris, D. 1994 G-snakes: Nonholonomic Kinematic Chains on Lie Groups. In *Proceedings of the 33rd IEEE Conference on Decision and Control (CDC)*, vol. 3, pp. 2955–2960. Lake Buena Vista, FL.
- Kudrolli, A., Lumay, G., Volfson, D. & Tsimring, L. S. 2008 Swarming and Swirling in Self-Propelled Polar Granular Rods. *Physical Review Letters*, **100**, 058 001. (doi:10.1103/PhysRevLett.100.058001)
- Leonard, N. E., Paley, D. A., Davis, R. E., Fratantoni, D. M., Lekien, F. & Zhang, F. 2010 Coordinated Control of an Underwater Glider Fleet in an Adaptive Ocean Sampling Field Experiment in Monterey Bay. *Journal of Field Robotics*, **27**(6), 718–740. (doi:10.1002/rob.20366)
- Li, K.-C. 1985 From Stein’s Unbiased Risk Estimates to the Method of Generalized Cross Validation. *The Annals of Statistics*, **13**(4), 1352–1377. (doi:10.1214/aos/1176349742)

- Li, Z., Liu, Y., Hayward, R., Zhang, J. & Cai, J. 2008 Knowledge-Based Power Line Detection for UAV Surveillance and Inspection Systems. In *Proceedings of the 23rd International Conference on Image and Vision Computing, New Zealand (IVCNZ)*, pp. 1–6. Lincoln, New Zealand. (doi:10.1109/IVCNZ.2008.4762118)
- Liberzon, D. 2011 *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton, NJ: Princeton University Press.
- Liu, Y. & Nejat, G. 2013 Robotic Urban Search and Rescue: A Survey from the Control Perspective. *Journal of Intelligent & Robotic Systems*, **72**(2), 147–165. (doi:10.1007/s10846-013-9822-x)
- Machado, L., Silva Leite, F. & Krakowski, K. 2010 Higher-order Smoothing Splines versus Least Squares Problems on Riemannian Manifolds. *Journal of Dynamical and Control Systems*, **16**(1), 121–148. (doi:10.1007/s10883-010-9080-1)
- Marques, C., Cristovao, J., Lima, P., Frazao, J., Ribeiro, I. & Ventura, R. 2006 RAPOSA: Semi-Autonomous Robot for Rescue Operations. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3988–3993. Beijing, China. (doi:10.1109/IROS.2006.281836)
- Marsden, J. E. & Ratiu, T. S. 2003 *Introduction to Mechanics and Symmetry*, vol. 17 of *Texts in Applied Mathematics*. New York, NY: Springer.
- Massey, J. L. 1990 Causality, Feedback and Directed Information. In *Proceedings of the International Symposium of Information Theory and its Application (ISITA-90)*, pp. 330–305. Waikiki, HI.
- McBeath, M. K., Shaffer, D. M. & Kaiser, M. K. 1995 How Baseball Outfielders Determine where to Run to Catch Fly Balls. *Science*, **268**(5210), 569–573. (doi:10.1126/science.7725104)

- Miller, N., Garnier, S., Hartnett, A. T. & Couzin, I. D. 2013 Both Information and Social Cohesion Determine Collective Decisions in Animal Groups. *Proceedings of the National Academy of Sciences*, **110**(13), 5263–5268. (doi:10.1073/pnas.1217513110)
- Mischiati, M., Lin, H. T., Herold, P., Imler, E., Olberg, R. & Leonardo, A. 2014 Internal Models Direct Dragonfly Interception Steering. *Nature*. (doi:10.1038/nature14045)
- Morrison, D. D., Riley, J. D. & Zancanaro, J. F. 1962 Multiple Shooting Method for Two-point Boundary Value Problems. *Communications of the ACM*, **5**(12), 613–614. (doi:10.1145/355580.369128)
- Nagy, M., Akos, Z., Biro, D. & Vicsek, T. 2010 Hierarchical Group Dynamics in Pigeon Flocks. *Nature*, **464**(7290), 890–893. (doi:10.1038/nature08891)
- Olberg, R. M., Worthington, A. H. & Venator, K. R. 2000 Prey Pursuit and Interception in Dragonflies. *Journal of Comparative Physiology A*, **186**(2), 155–162. (doi:10.1007/s003590050015)
- Olfati-Saber, R. 2006 Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory. *IEEE Transactions on Automatic Control*, **51**(3), 401–420. (doi:10.1109/TAC.2005.864190)
- P3-DX Pioneer P3-DX. <http://ter.ps/p3dx>. Accessed: 2014-09-07.
- Pinto, E., Santana, P. & Barata, J. 2013 On Collaborative Aerial and Surface Robots for Environmental Monitoring of Water Bodies. In *Technological Innovation for the Internet of Things* (eds L. Camarinha-Matos, S. Tomic & P. Graca), vol. 394 of *IFIP Advances in Information and Communication Technology*, pp. 183–191. Springer Berlin Heidelberg. (doi:10.1007/978-3-642-37291-9_20)

- Pontryagin, L. S., Boltyanskii, V. G., Gamkrelidze, R. V. & Mishchenko, E. F. 1962 *The Mathematical Theory of Optimal Processes*. Interscience Publishers, 1st edn.
- Quigley, M., Goodrich, M. A., Griffiths, S., Eldredge, A. & Beard, R. W. 2005 Target Acquisition, Localization, and Surveillance Using a Fixed-Wing Mini-UAV and Gimbaled Camera. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2600–2605. Barcelona, Spain. (doi:10.1109/ROBOT.2005.1570505)
- Reddy, P. V. 2007 *Steering Laws for Pursuit*. Master’s Thesis: University of Maryland, College Park.
- Reddy, P. V., Justh, E. W. & Krishnaprasad, P. S. 2006 Motion Camouflage in Three Dimensions. In *Proceedings of 45th IEEE Conference on Decision and Control (CDC)*, pp. 3327–3332. San Diego, CA. (doi:10.1109/CDC.2006.377777)
- Reddy, P. V., Justh, E. W. & Krishnaprasad, P. S. 2007 Motion camouflage with sensorimotor delay. In *Proceedings of the 46th IEEE Conference on Decision and Control (CDC)*, pp. 1660 –1665. New Orleans, LA. (doi:10.1109/CDC.2007.4434522)
- Ren, W. 2008 On Consensus Algorithms for Double-Integrator Dynamics. *IEEE Transactions on Automatic Control*, **53**(6), 1503–1509. (doi:10.1109/TAC.2008.924961)
- Ren, W. & Atkins, E. 2005 Second-order Consensus Protocols in Multiple Vehicle Systems with Local Interactions. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference (GNC)*. San Francisco, CA. (doi:10.2514/6.2005-6238)
- Reynolds, C. W. 1987 Flocks, herds and schools: A distributed behavioral model. *Computer Graphics*, **21**(4), 25–34. (doi:10.1145/37402.37406)

- Roberts, L., Bruck, H. & Gupta, S. K. 2014 Autonomous Loitering Control for a Flapping Wing Miniature Aerial Vehicle with Independent Wing Control. In *Proceedings of the ASME Mechanism and Robotics Conference*.
- ROS Robot Operating System (ROS). <http://www.ros.org>. Accessed: 2014-09-07.
- Savitzky, A. & Golay, M. J. E. 1964 Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry*, **36**(8), 1627–1639. (doi:10.1021/ac60214a047)
- Schafer, R. W. 2011 What Is a Savitzky-Golay Filter? [Lecture Notes]. *IEEE Signal Processing Magazine*, **28**(4), 111–117. (doi:10.1109/MSP.2011.941097)
- Sfakiotakis, M., Kazakidi, A., Chatzidaki, A., Evdaimon, T. & Tsakiris, D. P. 2014 Multi-arm Robotic Swimming with Octopus-Inspired Compliant Web. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- Shaikh, S. M. & Caines, P. E. 2007 On the Hybrid Optimal Control Problem: Theory and Algorithms. *IEEE Transactions on Automatic Control*, **52**(9), 1587–1603. (doi:10.1109/TAC.2007.904451)
- Shan, S., Egerstedt, M. & Martin, C. 2000 Control Theoretic Smoothing Splines. *IEEE Transactions on Automatic Control*, **45**(12), 2271–2279. (doi:10.1109/9.895563)
- Silverman, L. M. & Meadows, H. E. 1967 Controllability and Observability in Time-Variable Linear Systems. *SIAM Journal on Control and Optimization*, **5**(1), 64–73. (doi:10.1137/0305005)
- Simmons, J. A., Fenton, M. B. & O’Farrell, M. J. 1979 Echolocation and Pursuit of Prey by Bats. *Science*, **203**(4375), 16–21. (doi:10.1126/science.758674)

- Solo, V. 1996 A SURE-Fired Way to Choose Smoothing Parameters in Ill-Conditioned Inverse Problems. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, vol. 3, pp. 89–92. Lausanne, Switzerland. (doi:10.1109/ICIP.1996.560376)
- Spooner, G. M. 1931 Some Observations on Schooling in Fish. *Journal of the Marine Biological Association of the United Kingdom (New Series)*, **17**, 421–448. (doi:10.1017/S0025315400050943)
- Srinivasan, M. V. & Davey, M. 1995 Strategies for Active Camouflage of Motion. *Proceedings of the Royal Society B: Biological Sciences*, **259**(1354), 19–25. (doi:10.1098/rspb.1995.0004)
- Stein, C. M. 1981 Estimation of the Mean of a Multivariate Normal Distribution. *The Annals of Statistics*, **9**(6), 1135–1151. (doi:10.1214/aos/1176345632)
- Strandburg-Peshkin, A., Twomey, C. R., Bode, N. W. F., Kao, A. B., Katz, Y., Ioannou, C. C., Rosenthal, S. B., Torney, C. J., Wu, H. S. *et al.* 2013 Visual Sensory Networks and Effective Information Transfer in Animal Groups. *Current Biology*, **23**(17), R709–R711. (doi:10.1016/j.cub.2013.07.059)
- Sussmann, H. J. & Willems, J. C. 1997 300 Years of Optimal Control: from the Brachystochrone to the Maximum Principle. *IEEE Control Systems Magazine*, **17**(3), 32–44. (doi:10.1109/37.588098)
- Tan, X., Kim, D., Usher, N., Laboy, D., Jackson, J., Kapetanovic, A., Rapai, J., Sabadus, B. & Zhou, X. 2006 An Autonomous Robotic Fish for Mobile Sensing. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5424–5429. Beijing, China. (doi:10.1109/IROS.2006.282110)
- Taringoo, F. & Caines, P. E. 2013 On the Optimal Control of Hybrid Systems on Lie Groups and the Exponential Gradient HMP Algorithm. In *Proceedings of the*

- 52nd *IEEE Conference on Decision and Control (CDC)*, pp. 2653–2658. Florence, Italy. (doi:10.1109/CDC.2013.6760283)
- Tikhonov, A. N. 1963 Solution of Incorrectly Formulated Problems and the Regularization Method. *Soviet Mathematics (Translated)*, **4**, 1035–1038.
- Tits, A. L. 2013 Notes for ENEE 664: Optimal control.
- Todorov, E. & Jordan, M. 1998 Smoothness Maximization Along A Predefined Path Accurately Predicts The Speed Profiles of Complex Arm Movements. *Journal of Neurophysiology*, **80**(2), 696–714.
- Tokekar, P., Bhadauria, D., Studenski, A. & Isler, V. 2010 A Robotic System for Monitoring Carp in Minnesota Lakes. *Journal of Field Robotics*, **27**(6), 779–789. (doi:10.1002/rob.20364)
- Tokekar, P., Branson, E., Vander Hook, J. & Isler, V. 2013 Tracking Aquatic Invaders: Autonomous Robots for Monitoring Invasive Fish. *IEEE Robotics Automation Magazine*, **20**(3), 33–41. (doi:10.1109/MRA.2012.2220506)
- Topaz, C. M., Bertozzi, A. L. & Lewis, M. A. 2006 A Nonlocal Continuum Model for Biological Aggregation. *Bulletin of Mathematical Biology*, **68**(7), 1601–1623. (doi:10.1007/s11538-006-9088-6)
- Tsitsiklis, J. N., Bertsekas, D. P. & Athans, M. 1986 Distributed Asynchronous Deterministic and Stochastic Gradient Optimization Algorithms. *IEEE Transactions on Automatic Control*, **31**(9), 803–812. (doi:10.1109/TAC.1986.1104412)
- ViCoN Vicon Motion Capture System. <http://www.vicon.com>. Accessed: 2014-09-07.
- Vicsek, T., Czirók, A., Ben-Jacob, E., Cohen, I. & Shochet, O. 1995 Novel Type of

- Phase Transition in a System of Self-Driven Particles. *Physical Review Letters*, **75**, 1226–1229. (doi:10.1103/PhysRevLett.75.1226)
- Vicsek, T. & Zafeiris, A. 2012 Collective Motion. *Physics Reports*, **517**(3-4), 71–140. (doi:http://dx.doi.org/10.1016/j.physrep.2012.03.004)
- Wahba, G. 1990 *Spline Models for Observational Data*, vol. 59 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Philadelphia, PA: SIAM.
- Wahba, G. & Wold, S. 1975 A Completely Automatic French Curve: Fitting Spline Functions by Cross-Validation. *Communications in Statistics*, **4**(1), 1–17. (doi:10.1080/03610927508827223)
- Wei, E., Justh, E. W. & Krishnaprasad, P. S. 2009 Pursuit and an Evolutionary Game. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **465**(2105), 1539–1559. (doi:10.1098/rspa.2008.0480)
- Weimerskirch, H., Martin, J., Clerquin, Y., Alexandre, P. & Jiraskova, S. 2001 Energy Saving in Flight Formation. *Nature*, **413**(6857), 697 – 698. (doi:http://dx.doi.org/10.1038/35099670)
- Wright, G. S., Chiu, C., Xian, W., Wilkinson, G. S. & Moss, C. F. 2014 Social Calls Predict Foraging Success in Big Brown Bats. *Current Biology*, **24**(8), 885–889. (doi:10.1016/j.cub.2014.02.058)
- Young, G. F., Scardovi, L., Cavagna, A., Giardina, I. & Leonard, N. E. 2013 Starling Flock Networks Manage Uncertainty in Consensus at Low Cost. *PLoS Computational Biology*, **9**(1), e1002894. (doi:10.1371/journal.pcbi.1002894)
- YouTube-Video Implementation Videos. <http://ter.ps/flockvdo>. Accessed: 2014-09-30.

- Yu, W., Chen, G. & Cao, M. 2010 Some Necessary and Sufficient Conditions for Second-Order Consensus in Multi-Agent Dynamical Systems. *Automatica*, **46**(6), 1089 – 1095. (doi:10.1016/j.automatica.2010.03.006)
- Yu, W., Zheng, W. X., Chen, G., Ren, W. & Cao, M. 2011 Second-order Consensus in Multi-Agent Dynamical Systems with Sampled Position Data. *Automatica*, **47**(7), 1496 – 1503. (doi:10.1016/j.automatica.2011.02.027)
- Zhang, H. P., Beer, A., Florin, E. L. & Swinney, H. L. 2010 Collective Motion and Density Fluctuations in Bacterial Colonies. *Proceedings of the National Academy of Sciences*, **107**(31), 13 626–13 630. (doi:10.1073/pnas.1001651107)
- Zhang, Z., Tomlinson, J. & Martin, C. F. 1997 Splines and Linear Control Theory. *Acta Applicandae Mathematica*, **49**(1), 1–34. (doi:10.1023/A:1005856801842)
- Zhou, Y., Egerstedt, M. & Martin, C. 2005 Control Theoretic Splines with Deterministic and Random Data. In *Proceedings of 44th IEEE Conference on Decision and Control and 2005 European Control Conference (CDC-ECC)*, pp. 362–367. Seville, Spain. (doi:10.1109/CDC.2005.1582182)
- Zhou, Y., Egerstedt, M. & Martin, C. 2006 Hilbert Space Methods for Control Theoretic Splines: A Unified Treatment. *Communications in Information and Systems*, **6**(1), 55–82. (doi:10.4310/CIS.2006.v6.n1.a4)