

SYMMETRY ANALYSIS APPLIED
TO MULTI-AGENT SYSTEMS

A Thesis

Submitted to the Graduate School
of the University of Notre Dame
in Partial Fulfillment of the Requirements
for the Degree of

Master of Science

in

Mechanical Engineering

by

Ashley M. Nettleman

Bill Goodwine, Director

Graduate Program in Aerospace and Mechanical Engineering

Notre Dame, Indiana

November 2014

© Copyright by
Ashley Nettleman
2014
All Rights Reserved

SYMMETRY ANALYSIS APPLIED
TO MULTI-AGENT SYSTEMS

Abstract

by

Ashley M. Nettleman

The multi-agent system of interest is a planar formation control problem, where each agent references a pre-specified number of agents. Since the formation control law does not specify the location and orientation of the formation in planar space, continuous symmetries arise. Through a previously established methodology, it is possible to calculate the continuous symmetries for a system of second order differential equations. The symmetries were found for a specific neighbor graph and were extended to the general case. The symmetries associated with planar motion were used to define coordinate transformations that reduced the system of interest to one in which the origin is the set of all possible formations. It is now possible to perform stability analysis of the formation by studying the stability properties of the origin of the reduced system. This will be beneficial for showing extended stability properties, like boundedness.

This thesis is dedicated to my loving husband, Joseph Nettleman, who has encouraged and supported me during my graduate studies. Thank you for putting up with my procrastination, long work hours late into the night, and listening to me ramble on about my research all while working towards your medical degree.

CONTENTS

FIGURES	v
ACKNOWLEDGMENTS	vi
SYMBOLS	vii
CHAPTER 1: INTRODUCTION	1
CHAPTER 2: BACKGROUND THEORY ON NETWORKS AND SYMMETRIES	3
2.1 System of interest and network terminology	3
2.2 Definition of a symmetry	6
2.3 Prolongation of the generator for use with differential equations	9
2.4 Using a general form to find the symmetries for a system	10
CHAPTER 3: FINDING THE SYMMETRIES	13
3.1 General theory for a system of second order differential equations	13
3.2 Applying the theory to the system of interest	14
3.3 The symmetries for the system of interest	17
3.4 Verification of the symmetries	20
3.5 Special case for three agents	23
CHAPTER 4: USING THE SYMMETRIES FOR REDUCTION	24
4.1 General theory and notation	24
4.2 Two-agent example	25
4.2.1 Translation in x -direction symmetry	26
4.2.2 Translation in y -direction symmetry	27
4.2.3 A comment on the coordinate transformations thus far	29
4.2.4 Rotational symmetry	30
4.3 Five-agent example	32
4.4 Extension of method for N agents	40
CHAPTER 5: STABILITY ANALYSIS	44
5.1 Stability analysis for the original system	44
5.2 Lyapunov stability for the reduced two-agent system	45

CHAPTER 6: CONCLUSIONS AND FUTURE WORK	47
APPENDIX A: MATHEMATICA PROGRAM USED TO SOLVE THE SYM- METRY EQUATION	49
BIBLIOGRAPHY	55

FIGURES

2.1	Examples of a (a) connected undirected network, (b) weakly connected directed network, and (c),(d) two strongly connected directed graphs	5
2.2	Examples of a (a) disconnected undirected network and (b) connected undirected network	5
2.3	The two possible configurations for a rigid and foldable neighbor graph	6
2.4	Two examples of neighbor graphs that are rigid and not foldable. . .	7
4.1	Neighbor graphs for the two five agent cases	33

ACKNOWLEDGMENTS

I would like to acknowledge the support of my mother, Lori Kulczycki, and my mother-in-law, Kathy Nettleman, for their support and invaluable assistance in editing my thesis. It still amazes me the errors they were able to find, even though they understood very little about my research.

I would also like to acknowledge my father, Jerome Kulczycki, and my father-in-law, Bob Nettleman, for their continued enthusiastic support of my career path.

The editorial skills of Galen Scott are also acknowledged.

The support of the US National Science Foundation under the CPS Large Grant No. CNS-1035655 is gratefully acknowledged.

SYMBOLS

- A partial differential equation form of the differential equation
- a index, typically for the symmetry condition equations
- B matrix term in the symmetry result
- b index for summation
- C matrix term in the symmetry result
- c index for summation
- D matrix term in the symmetry result for three agents or
the derivative function in Mathematica code
- d_{ij} error in the distance between agents i and j
- \hat{d}_{ij} desired distance between agents i and j
- d_j^i error in the distance between agent specified by q^i and the agent an index
distance of j away
- \hat{d}_j^i desired distance between agent specified by i and the agent an index
distance of j away
- d_1 error in distance between the two agents in the two agent case
- d total derivative
- f a function
- g^i position based terms in the control law for the differential equation
corresponding to q^i
- H^i differential equation corresponding to q^i set equal to zero,
$$H^i = \ddot{q}^i - \omega^i = 0$$
- H refers to the whole set of H^i terms or a general case

- i index
- j index typically used for a neighboring agent or the index distance for a neighbor
- k index typically used for the corresponding x or y term such that (q^i, q^k) are the dynamics for an agent
- l index for neighboring agent coordinates
- m index for neighboring agent coordinates or the number of constraints on a formation
- N number of agents for the system of interest
- \mathcal{N}_i the neighborhood for an agent i
- $\tilde{\mathcal{N}}_i$ the neighborhood for the agent specified by q^i
- n order of a differential equation
- \mathcal{O} order of magnitude
- p^i quadrature defined to reduce the order of the system
- q_j^i dynamics for the system, where i indices the agent and j is used to indicate the step in the coordinate reduction
- q^i dynamics for the system, where i indices the agent
- q dependent variable
- \bar{q} column vector of the q^i terms in the symmetry result
- r^i coordinate transformation variable corresponding to $Xr^i = 0$
- r coordinate transformation variable corresponding to $Xr = 0$
- s^i coordinate transformation variable corresponding to $Xs^i = 1$
- s coordinate transformation variable corresponding to $Xs = 1$
- t independent variable, typically time
- V_i Lyapunov function corresponding to agent i
- V Summation of the individual Lyapunov functions, V_i
- X_j^i linear operator form for the j th coordinate reduction on the i th

- individual symmetry
- X^i linear operator form for the i th individual symmetry
- X linear operator form for the symmetry and typically the general form with free variables
- x_i the position on the x -axis for agent i
- x the position on the x -axis
- \tilde{x} the position on the x -axis after a point transformation
- $y^{(n)}$ the n th derivative of y with respect to time
- y_i the position on the y -axis for agent i
- y the position on the y -axis
- \tilde{y} the position on the y -axis after a point transformation
- $\alpha^0(t)$ function depending only on time that is used in finding the ξ term
- α^0 constant that is used in finding the ξ term
- $\alpha^i(t)$ function of time that has a correspondence with q^i that is used in finding the ξ term
- α^i constant that has a correspondence with q^i that is used in finding the ξ term
- α index
- $\beta^i(t)$ function of time that has a correspondence with the i th equation that is used in the η term
- β_j^i constant that has a correspondence with the i th equation and the q^j term that is used in the η term
- $\overline{\beta(t)}$ column vector of the $\beta^i(t)$ terms in the symmetry result
- $\overline{\Gamma}$ matrix term in the symmetry result
- $\gamma_j^i(t)$ function of time that corresponds to the i th equation and the q^j term
- γ_j^i constant that corresponds to the i th equation and the q^j term
- δ_{ab} Kronecker delta function that is equal to one when $a = b$ and is equal to

	zero otherwise
∂	partial derivative
ε	variation parameter in the point transformation corresponding to a symmetry
η_k^i	spatial coefficient term corresponding to the q^i term in the k th symmetry
η^i	spatial coefficient term corresponding to the q^i term
η	spatial coefficient term in the general symmetry
$\bar{\eta}$	column vector of the η^i terms in the symmetry result
λ_{max}	maximum eigenvalue
λ	a typically nonconstant factor or a trajectory of the solution
ξ	time coefficient term in the general symmetry
ϕ^α	the complete set of solutions to $y^{(n)} = \omega$
ϕ_i^{-1}	the inverse of the i th coordinate transformation
ϕ_i	the i th coordinate transformation
Ω^i	a function of the solutions ϕ^α
ω^i	control law corresponding to q^i such that $H^i = \ddot{q}^i - \omega^i = 0$
ω	general form for the control law
∞	infinity
\int_0^∞	integral from zero to infinity
$,_i$	partial derivative with respect to q^i
$,_q$	partial derivative with respect to q
$,_t$	partial derivative with respect to t

CHAPTER 1

INTRODUCTION

The study of multi-agent systems has typically been fraught with the difficulty of dealing with the resulting complex system. Add in varying interconnections between the agents, and the analysis of such a system quickly becomes difficult. However, usually these systems have several symmetric properties, since they tend to have very similar, if not identical agents. Therefore, it is of interest to be able to extract the symmetric information and work instead with a reduced system.

Unfortunately, using the symmetries to reduce the system becomes complicated by the fact that there are several types of symmetries. Typically the first type of symmetry that comes to mind is the discrete reflection, or mirror, symmetry. Objects with this symmetry generally have an obvious centerline where the features on one side of the centerline are just a reflection, or mirror, of the other side. There also exist continuous symmetries, where by varying a parameter it is possible to go from one solution to another. Of particular interest are the continuous symmetries that return the original solution when the varying parameter is equal to zero and the transformation defined by the parameter is continuous. Some examples of these continuous symmetries include translation and rotation. These are the symmetries that will be explored in attempts to reduce the complexity of the dynamics of a multi-agent system.

Multi-agent systems have been studied by many others [2, 3, 14, 16, 17, 19]. Of particular interest is being able to guarantee that the system of agents, or robots, is able to make it to a final destination or formation. It is also desirable that the agents

are able to perform as intended with minimal external control input, as this can allow for reduced required computing power. With this in mind, some researchers [11, 13] were able to design control laws that created imaginary agents that acted as guides to help avoid obstacles. Discrete symmetries have also been studied [5–7, 9] in an effort to reduce the complexity of the system and further study the stability of the formation, or the ability of the agents to achieve the desired formation.

Unfortunately, there is a common potential pitfall with some of the prior work. Typically LaSalle’s Invariance Principle has been used to show that the system is stable; however, it requires that a compact invariant set be defined [10]. A compact set requires that the area in which one is working be closed and bounded. The majority of the time, this is not an issue; however, with the scale of the systems becoming larger and the range of motion becoming unbounded, this is quickly becoming an important aspect to investigate. Therefore, it is of interest to be able to reduce and redefine the system, such that the control laws have the origin as the set of all possible formations for the system of agents. This allows for the use of Lyapunov’s Theorem, which does not require a compact set to be defined. Additionally, numerous extensions have been derived to be able to show boundedness, or the ability of the system to remain stable with disturbances. Since disturbances exist in the real world, it is of added benefit to be able to use these extensions to show boundedness when the system is not ideal.

The organization of this work is such that the process for reducing the system of interest is broken down into steps. In Chapter 2, some basic definitions and preliminaries of networks of agents and symmetries are introduced along with the system of interest. The method to find the symmetries for a system of second order differential equations is presented in Chapter 3, along with the results for the system of interest. These symmetries are then used to reduce the system of interest in Chapter 4, and stability analysis is shown for the reduced dynamics found for the two agent case in Chapter 5.

CHAPTER 2

BACKGROUND THEORY ON NETWORKS AND SYMMETRIES

2.1 System of interest and network terminology

The system of interest is an interconnected system of agents with a control law driving the system to a formation. These agents can be robots, vehicles, a representation of a cyber-physical system, or any system in which the dynamics can be written as

$$\begin{aligned}\ddot{x}_i &= -\dot{x}_i - \sum_{j \in \mathcal{N}_i} (x_i - x_j) d_{ij} \\ \ddot{y}_i &= -\dot{y}_i - \sum_{j \in \mathcal{N}_i} (y_i - y_j) d_{ij},\end{aligned}$$

with

$$d_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2 - \hat{d}_{ij},$$

where \mathcal{N}_i is the set the indices corresponding to the agents that agent i references, \hat{d}_{ij} is the square of the desired distance between agents i and j , and i ranges from one to N , which is the number of agents in the system.

For ease of analysis, the equations will be written in terms of one coordinate q , where

$$x_i = q^{2i-1} \quad \text{and} \quad y_i = q^{2i} \quad \text{for} \quad i \in \{1, \dots, N\}.$$

The change from subscripts to superscripts is done to match the notation used in

[18]. The system of interest is then

$$\ddot{q}^i = \omega^i = -\dot{q}^i - \sum_{j \in \tilde{\mathcal{N}}_i} d_j^i (q^i - q^l)$$

$$d_j^i = (q^i - q^l)^2 + (q^k - q^m)^2 - \hat{d}_j^i,$$

where $i \in \{1, \dots, 2N\}$, $k = i - (-1)^i$, $l = i + 2j \pmod{2N}$, $m = k + 2j \pmod{2N}$, N is the number of agents, $\tilde{\mathcal{N}}_i$ is the set of neighbors with which agent i communicates, and \hat{d}_j^i is the desired distance between agent i and its neighbor that is j away.

The set of neighbors that an agent references, or communicates with, can be represented as a network. This network will be referred to as the neighbor graph for the system. The nodes of the neighbor graph are the agents, and an edge exists if two agents communicate with each other. These edges can be directed or undirected, depending on the type of communication between the agents. If the communication is defined as being one-directional, then the edges will be directed. Otherwise, the edges will be undirected. Depending on whether the network is directed or undirected, [12] defines two different standards of connectivity. For an undirected network, the network is connected if it is possible to travel from any node to another by traveling along the edges. Figure 2.1a shows an example of a connected undirected network. On the other hand, a directed network can be either strongly connected, weakly connected, or disconnected. A strongly connected network is one in which it is possible to travel from any node to any other by traveling in the direction of the edges. If this is not possible, but if the directions on the edges are removed and the resulting undirected network is connected, then the directed network is weakly connected. Figure 2.1b shows a weakly connected directed network, while Figures 2.1c and 2.1d are both strongly connected directed graphs. An example of a disconnected undirected network is shown in Figure 2.2a, along with a connected undirected network with the same number of nodes in Figure 2.2b.

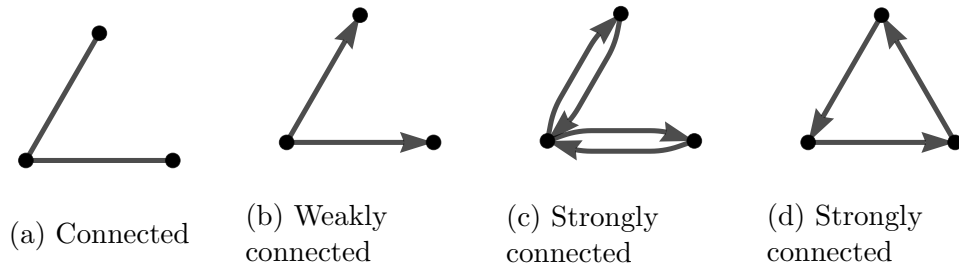


Figure 2.1. Examples of a (a) connected undirected network, (b) weakly connected directed network, and (c),(d) two strongly connected directed graphs

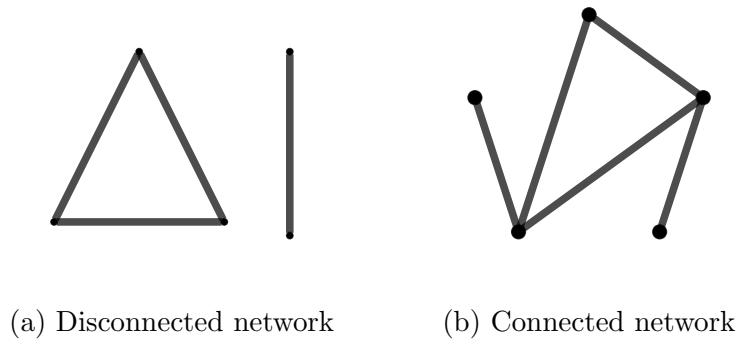


Figure 2.2. Examples of a (a) disconnected undirected network and (b) connected undirected network

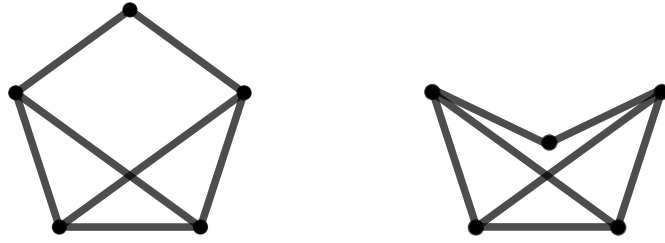


Figure 2.3. The two possible configurations for a rigid and foldable neighbor graph

For formation theory, [14] defines two additional network concepts: rigidity and foldability. A formation's neighbor graph is rigid if there exists at least $2N - 3$ edges. Even if a neighbor graph is rigid, multiple formations may still be possible through a discrete transformation. When this is possible, the neighbor graph is foldable. In order to define a formation that is rigid and not foldable, each agent must have edges connecting it to three other agents. Figure 2.2b is an example of a five-agent neighbor graph that is not rigid. The two different possible configurations for a rigid foldable neighbor graph are shown in Figure 2.3. Note that neighbor graphs do not necessarily have to appear the same as the formation; however, for illustrative purposes, the neighbor graph is shown to have edge lengths corresponding to the desired distances between agents. Examples of rigid and not foldable neighbor graphs are shown in Figure 2.4.

2.2 Definition of a symmetry

In order to factor out the symmetries present in a system, one must first be able to find and represent the symmetries. It is possible to represent continuous Lie symmetries by an infinitesimal point transformation in terms of at least one parameter. For example, the rotational symmetry in the plane can be represented

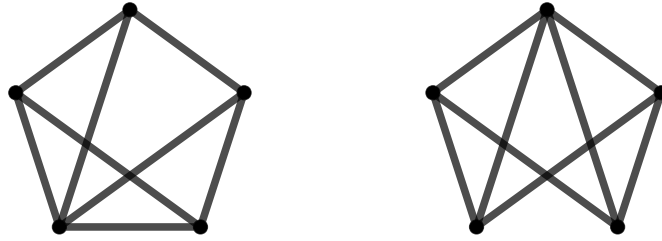


Figure 2.4. Two examples of neighbor graphs that are rigid and not foldable.

by the point transformation

$$\begin{aligned}\tilde{x}(x, y; \varepsilon) &= x \cos \varepsilon - y \sin \varepsilon \\ \tilde{y}(x, y; \varepsilon) &= x \sin \varepsilon + y \cos \varepsilon,\end{aligned}$$

where ε can be thought of as the angle of rotation. A function, $f(x, y) = 0$ has a rotational symmetry if $f(x, y) = f(\tilde{x}, \tilde{y}) = 0$ for the point transformation defined above. In other words, the point transformation is able to take one solution of $f(x, y) = 0$ to another solution of $f(x, y) = 0$.

It is also possible to represent the symmetry as a linear operator, called a generator and denoted by X . The generator can be found from the point transformation, or a general form can be assumed to find all of the symmetries of a system. The generator can be defined by terms in the Taylor series expansion of the point transformation

about $\varepsilon = 0$. Consider

$$\begin{aligned}
\tilde{x}(x, y; \varepsilon) &= \tilde{x}(x, y; 0) + \left. \frac{\partial \tilde{x}}{\partial \varepsilon} \right|_{\varepsilon=0} (\varepsilon - 0) + \dots \\
&= x + \varepsilon \eta_1(x, y) + \dots \\
&= x + \varepsilon X x + \dots \\
\tilde{y}(x, y; \varepsilon) &= \tilde{y}(x, y; 0) + \left. \frac{\partial \tilde{y}}{\partial \varepsilon} \right|_{\varepsilon=0} (\varepsilon - 0) + \dots \\
&= y + \varepsilon \eta_2(x, y) + \dots \\
&= y + \varepsilon X y + \dots,
\end{aligned}$$

which gives

$$X = \eta_1(x, y) \frac{\partial}{\partial x} + \eta_2(x, y) \frac{\partial}{\partial y}.$$

Note that is also possible to have a transformation in the independent variable. When this is the case, the general expression for the symmetry is

$$X = \xi(t, x, y) \frac{\partial}{\partial t} + \eta_1(t, x, y) \frac{\partial}{\partial x} + \eta_2(t, x, y) \frac{\partial}{\partial y}.$$

The check for whether a function, $f(x, y) = 0$, has the symmetry defined by a generator X is simply $Xf = 0$, which is satisfied if the action of the symmetry leaves the function invariant. For example, the rotational symmetry has the generator

$$X = -y \frac{\partial}{\partial x} + x \frac{\partial}{\partial y},$$

which when acting on the function $f(x, y) = x^2 + y^2 - 4 = 0$, leaves the function invariant. In other words,

$$Xf = \left(-y \frac{\partial}{\partial x} + x \frac{\partial}{\partial y} \right) (x^2 + y^2 - 4) = -y(2x) + x(2y) = 0.$$

2.3 Prolongation of the generator for use with differential equations

In order to apply the generator to differential equations, one must first prolong the generator to include derivative terms. This is done by taking the Taylor series expansion of the transformations for the derivatives. For an independent variable t , the Taylor series expansion for the x derivative is

$$\begin{aligned} \frac{d\tilde{x}}{d\tilde{t}}(t, x, y; \varepsilon) &= \frac{d\tilde{x}}{d\tilde{t}}(t, x, y; 0) + \frac{\partial}{\partial \varepsilon} \left(\frac{d\tilde{x}}{d\tilde{t}} \right) \Big|_{\varepsilon=0} (\varepsilon - 0) + \dots \\ &= \frac{dx}{dt} + \varepsilon \eta'_1(t, x, y, dx/dt, dy/dt) + \dots \\ &= \frac{dx}{dt} + \varepsilon X \frac{dx}{dt} + \dots, \end{aligned}$$

where

$$\begin{aligned} \eta'_1(t, x, y, dx/dt, dy/dt) &= \frac{\partial}{\partial \varepsilon} \left(\frac{d\tilde{x}}{d\tilde{t}} \right) \Big|_{\varepsilon=0} \\ &= \frac{\partial}{\partial \varepsilon} (d\tilde{x} d\tilde{t}^{-1}) \Big|_{\varepsilon=0} \\ &= \left(\frac{\partial}{\partial \varepsilon} (d\tilde{x}) d\tilde{t}^{-1} + (-1) d\tilde{x} \frac{\partial}{\partial \varepsilon} (d\tilde{t}) d\tilde{t}^{-2} \right) \Big|_{\varepsilon=0} \\ &= \left(d \left(\frac{\partial \tilde{x}}{\partial \varepsilon} \right) d\tilde{t}^{-1} - d \left(\frac{\partial \tilde{t}}{\partial \varepsilon} \right) d\tilde{x} d\tilde{t}^{-2} \right) \Big|_{\varepsilon=0} \\ &= \left(\frac{d}{d\tilde{t}} (\eta_1) - \frac{d}{d\tilde{t}} (\xi) \frac{d\tilde{x}}{d\tilde{t}} \right) \Big|_{\varepsilon=0} \\ &= \frac{d\eta_1}{dt} - \frac{d\xi}{dt} \frac{dx}{dt}. \end{aligned}$$

Since there is a transformation in the independent variable, η'_1 does not necessarily equal $d\eta_1/dt$. However, η'_1 is equal to $d\eta_1/dt$ when ξ is equal to a constant or zero.

The expression for the prolongation of the generator is then

$$X = \xi \frac{\partial}{\partial t} + \eta_1 \frac{\partial}{\partial x} + \eta_2 \frac{\partial}{\partial y} + \eta'_1 \frac{\partial}{\partial \dot{x}} + \eta'_2 \frac{\partial}{\partial \dot{y}} + \eta''_1 \frac{\partial}{\partial \ddot{x}} + \eta''_2 \frac{\partial}{\partial \ddot{y}} + \dots,$$

where $\dot{x} = \frac{dx}{dt}$, $\dot{y} = \frac{dy}{dt}$ and

$$\begin{aligned}\eta'_1 &= \frac{d\eta_1}{dt} - \dot{x} \frac{d\xi}{dt} & \eta_1^{(i)} &= \frac{d\eta_1^{(i-1)}}{dt} - \frac{d^i x}{dt^i} \frac{d\xi}{dt} \\ \eta'_2 &= \frac{d\eta_2}{dt} - \dot{y} \frac{d\xi}{dt} & \eta_2^{(i)} &= \frac{d\eta_2^{(i-1)}}{dt} - \frac{d^i y}{dt^i} \frac{d\xi}{dt}.\end{aligned}$$

Note that if ξ is equal to a constant, which corresponds to the independent variable being transformed by a constant shift, $\eta_j^{(i)} = \frac{d^i \eta_j}{dt^i}$. In other words, the only time when $\eta_j^{(i)}$ is equal to the i th derivative of η_j is when the independent variable is subjected to at most a constant shift. For further details on the prolongation of the symmetry, see [18].

The prolongation of the rotational symmetry is

$$X = -y \frac{\partial}{\partial x} + x \frac{\partial}{\partial y} - \dot{y} \frac{\partial}{\partial \dot{x}} + \dot{x} \frac{\partial}{\partial \dot{y}} - \ddot{y} \frac{\partial}{\partial \ddot{x}} + \ddot{x} \frac{\partial}{\partial \ddot{y}},$$

where the independent variable is t .

A differential equation, $H = 0$, has a symmetry defined by a generator X if $XH = 0$. Note that for systems of differential equations, $H^1 = 0, H^2 = 0$, it is possible that $XH^1 = H^2$, which equals zero since H^2 equals zero by definition. Since $H = 0$ and $XH = 0$, the generator X maps solutions of $H = 0$ into another solution of $H = 0$.

2.4 Using a general form to find the symmetries for a system

One way to find the symmetries of a differential equation is by “solving for X .” This is done systematically when the differential equation is expressed as a linear operator. For simplicity, this will be shown for one dependent variable y and one independent variable t . The relationship between a differential equation $y^{(n)} = \omega$

and its linear operator is found by the first integrals f of a differential equation and is

$$\left(\frac{\partial}{\partial t} + y' \frac{\partial}{\partial y} + \cdots + \omega \frac{\partial}{\partial y^{(n-1)}} \right) f = Af = 0,$$

where $y^{(i)}$ denotes the i th derivative of y with respect to the independent variable t . Note that a system of differential equations can be represented by a single linear operator. This will be shown in Chapter 3.

The general expression for a symmetry of this system is

$$X = \xi \frac{\partial}{\partial t} + \eta \frac{\partial}{\partial y} + \eta' \frac{\partial}{\partial y'} + \cdots + \eta^{(n-1)} \frac{\partial}{\partial y^{(n-1)}},$$

where $\eta' = \frac{d\eta}{dt} - y' \frac{d\xi}{dt}$, $\eta^{(i)} = \frac{d\eta_i}{dt} - y^{(n)} \frac{d\xi}{dt}$. Let ϕ^α be the complete set of n independent solutions of $y^{(n)} = \omega$. Since X is a symmetry of this system, $X\phi^\alpha$ must also be a solution since the generator maps solutions into solutions. The relationships between X and A with the solutions ϕ^α are

$$X\phi^\alpha = \Omega^\alpha(\phi^\beta), \quad A\phi^\alpha = 0 = A\Omega^\alpha,$$

where Ω^α is defined to be a function of the solutions such that it is also a solution to $y^{(n)} = \omega$.

By use of the skew symmetric commutator, $[X, A] = XA - AX$, it is possible to eliminate the unknown functions Ω^α . Note that $[X, A]$ is also a linear operator and that

$$[X, A]\phi^\alpha = X(A\phi^\alpha) - A(X\phi^\alpha) = X(0) - A(\Omega^\alpha) = 0 - 0 = 0.$$

This means that $[X, A]f = 0$ has the same set of solutions as $Af = 0$ since this is valid for all of the solutions ϕ^α . Therefore, the linear operator $[X, A]$ can only differ

from the linear operator A by a factor λ ,

$$[X, A] = \lambda(t, y, y', \dots, y^{(n-1)})A.$$

By equating the coefficients of the partial derivatives, $\partial/\partial t, \partial/\partial y, \partial/\partial y', \dots$, in this equation, it is possible to solve for the unknowns ξ and η to determine the symmetries for the differential equation $y^{(n)} = \omega$. Further details on the terminology, method, and analysis can be found in [18] and [15].

CHAPTER 3

FINDING THE SYMMETRIES

3.1 General theory for a system of second order differential equations

A system of m second order differential equations

$$\ddot{q}^1 = \omega^1, \dots, \ddot{q}^m = \omega^m,$$

can be written as a partial differential equation $Af = 0$, where

$$A = \frac{\partial}{\partial t} + \dot{q}^1 \frac{\partial}{\partial q^1} + \omega^1 \frac{\partial}{\partial \dot{q}^1} + \dots + \dot{q}^m \frac{\partial}{\partial q^m} + \omega^m \frac{\partial}{\partial \dot{q}^m}.$$

The general form for a symmetry X of m second order differential equations is

$$X = \xi(t, q^1, \dots, q^m) \frac{\partial}{\partial t} + \eta^1(t, q^1, \dots, q^m) \frac{\partial}{\partial q^1} + \dots + \eta^m(t, q^1, \dots, q^m) \frac{\partial}{\partial q^m}.$$

From [18], the symmetry condition $[X, A] = \lambda A$ simplifies to (with $_{,t} = \partial/\partial t$, and $_{,c} = \partial/\partial q^c$)

$$\begin{aligned} & \xi \omega_{,t}^a + \eta^b \omega_{,b}^a + (\eta_{,t}^b + \dot{q}^c \eta_{,c}^b - \dot{q}^b \xi_{,t} - \dot{q}^b \dot{q}^c \xi_{,c}) \frac{\partial \omega^a}{\partial \dot{q}^b} + 2\omega^a (\xi_{,t} + \dot{q}^b \xi_{,b}) \\ & + \omega^b (\dot{q}^a \xi_{,b} - \eta_{,b}^a) + \dot{q}^a \dot{q}^b \dot{q}^c \xi_{,bc} + 2\dot{q}^a \dot{q}^c \xi_{,tc} - \dot{q}^c \dot{q}^b \eta_{,bc}^a + \dot{q}^a \xi_{,tt} - 2\dot{q}^b \eta_{,tb}^a - \eta_{,tt}^a = 0, \end{aligned}$$

with $(a, b, c = 1, \dots, m)$. Note that a is an index for the m equations, while b and c are summation indices. Recall that ξ and η^a are functions of t and q^a and not \dot{q}^a . This allows for the m symmetry condition equations to be split apart further, which

reduces the complexity of solving for ξ and η^a . The next step will be to split the equations apart based on the order of the \dot{q}^a terms.

3.2 Applying the theory to the system of interest

The system of interest is

$$\ddot{q}^i = \omega^i = -\dot{q}^i - \sum_{j \in \tilde{\mathcal{N}}_i} d_j^i (q^i - q^j)$$

$$d_j^i = (q^i - q^j)^2 + (q^k - q^m)^2 - \hat{d}_j^i,$$

where $i \in \{0, \dots, 2N\}$, $k = i - (-1)^i$, $l = i + 2j \pmod{2N}$, $m = k + 2j \pmod{2N}$, N is the number of agents, $\tilde{\mathcal{N}}_i$ is the set of neighbors with which agent i communicates, and \hat{d}_j^i is the desired distance between agent i and its neighbor that is j away. The control law ω^i is comprised of a damping term \dot{q}^i and a formation force that drives the system to a desired formation. Note that the system does not depend explicitly on time and the \dot{q}^i terms do not appear in the formation force. Therefore, for simplicity of notation, the formation force will be written as $g^a(q^1, \dots, q^{2N})$. In other words,

$$\ddot{q}^i = \omega^i = -\dot{q}^i - g^i.$$

The $m = 2N$ equations that define the symmetry are then

$$\eta^b(-g^a)_{,b} + (\eta_{,t}^b + \dot{q}^c \eta_{,c}^b - \dot{q}^b \xi_{,t} - \dot{q}^b \dot{q}^c \xi_{,c}) (-1) \delta_{ab} + 2(-\dot{q}^a - g^a) (\xi_{,t} + \dot{q}^b \xi_{,b})$$

$$+ (-\dot{q}^b - g^b) (\dot{q}^a \xi_{,b} - \eta_{,b}^a) + \dot{q}^a \dot{q}^b \dot{q}^c \xi_{,bc} + 2\dot{q}^a \dot{q}^c \xi_{,tc} - \dot{q}^c \dot{q}^b \eta_{,bc}^a + \dot{q}^a \xi_{,tt} - 2\dot{q}^b \eta_{,tb}^a - \eta_{,tt}^a = 0,$$

with $\delta_{ab} = 1$ if $a = b$ and 0 otherwise. Rewriting this system by decreasing powers of \dot{q}^a gives

$$\begin{aligned} & \dot{q}^a \dot{q}^b \dot{q}^c \xi_{,bc} + \dot{q}^b \dot{q}^c \xi_{,c} \delta_{ab} - 3\dot{q}^a \dot{q}^b \xi_{,b} + 2\dot{q}^a \dot{q}^c \xi_{,tc} - \dot{q}^c \dot{q}^b \eta_{,bc}^a \\ & - \dot{q}^c \eta_{,c}^b \delta_{ab} + \dot{q}^b \xi_{,t} \delta_{ab} - 2\dot{q}^a \xi_{,t} - 2\dot{q}^b g^a \xi_{,b} + \dot{q}^b \eta_{,b}^a - \dot{q}^a g^b \xi_{,b} + \dot{q}^a \xi_{,tt} - 2\dot{q}^b \eta_{,tb}^a \\ & - \eta_{,b}^a g_{,b}^a - \eta_{,t}^b \delta_{ab} - 2g^a \xi_{,t} + g^b \eta_{,b}^a - \eta_{,tt}^a = 0. \end{aligned}$$

Since the symmetry condition equations must hold for all values of \dot{q}^a and q^a , the coefficients for each order of \dot{q} and q^a must equal zero. Taking $\dot{q}^a \dot{q}^b \dot{q}^c \xi_{,bc} = 0$ yields

$$\xi = \alpha^0(t) + \alpha^1(t)q^1 + \cdots + \alpha^{2N}(t)q^{2N}.$$

Note that this means that

$$\begin{aligned} \xi_{,t} &= \dot{\alpha}^0(t) + \dot{\alpha}^1(t)q^1 + \cdots + \dot{\alpha}^{2N}(t)q^{2N} \\ \xi_{,b} &= \alpha^b(t) \\ \xi_{,tb} &= \dot{\alpha}^b(t). \end{aligned}$$

The symmetry condition equations are now

$$\begin{aligned} & \dot{q}^b \dot{q}^c \alpha^c(t) \delta_{ab} - 3\dot{q}^a \dot{q}^b \alpha^b(t) + 2\dot{q}^a \dot{q}^c \dot{\alpha}^c(t) - \dot{q}^c \dot{q}^b \eta_{,bc}^a \\ & - \dot{q}^c \eta_{,c}^b \delta_{ab} + \dot{q}^b \delta_{ab} (\dot{\alpha}^0(t) + \dot{\alpha}^c(t)q^c) - 2\dot{q}^a (\dot{\alpha}^0(t) + \dot{\alpha}^c(t)q^c) - 2\dot{q}^b g^a \alpha^b(t) \\ & + \dot{q}^b \eta_{,b}^a - \dot{q}^a g^b \alpha^b(t) + \dot{q}^a (\ddot{\alpha}^0(t) + \ddot{\alpha}^c(t)q^c) - 2\dot{q}^b \eta_{,tb}^a \\ & - \eta_{,b}^a g_{,b}^a - \eta_{,t}^b \delta_{ab} - 2g^a (\dot{\alpha}^0(t) + \dot{\alpha}^c(t)q^c) + g^b \eta_{,b}^a - \eta_{,tt}^a = 0. \end{aligned}$$

The next equation to solve is

$$\dot{q}^b \dot{q}^c \alpha^c(t) \delta_{ab} - 3\dot{q}^a \dot{q}^b \alpha^b(t) + 2\dot{q}^a \dot{q}^c \dot{\alpha}^c(t) - \dot{q}^c \dot{q}^b \eta_{,bc}^a = 0,$$

which simplifies to

$$2\dot{q}^a \dot{q}^b (\dot{\alpha}^b(t) - \alpha^b(t)) - \dot{q}^c \dot{q}^b \eta_{,bc}^a = 0.$$

Recall that a designates the symmetry condition equation, while b and c are summation indices. This gives that

$$\begin{aligned} 2(\dot{\alpha}^b(t) - \alpha^b(t)) &= \eta_{,ba}^a, \\ \eta_{,bc}^a &= 0 \quad \text{when } c \neq a. \end{aligned}$$

Since partial derivatives commute,

$$\begin{aligned} \eta_{,bc}^a &= 0, \\ \eta^a &= \beta^a(t) + \gamma_b^a(t) q^b \\ 2(\dot{\alpha}^b(t) - \alpha^b(t)) &= 0, \\ \dot{\alpha}^b(t) &= \alpha^b(t), \\ \alpha^b(t) &= \alpha^b \exp(t), \end{aligned}$$

where α^b is a scalar. The symmetry condition equations are now

$$\begin{aligned} &-\dot{q}^a \dot{\alpha}^0(t) + \dot{q}^a \ddot{\alpha}^0(t) - 2\dot{q}^b \dot{\gamma}_b^a(t) - 2\dot{q}^b g^a \alpha^b \exp(t) - \dot{q}^a g^b \alpha^b \exp(t) \\ &- g_{,b}^a \beta^b(t) - g_{,b}^a \gamma_c^b(t) q^c - \dot{\gamma}_b^a(t) q^b - 2g^a \dot{\alpha}^0(t) - 2g^a \alpha^b \exp(t) q^b + g^b \gamma_b^a(t) - \ddot{\gamma}_b^a(t) q^b \\ &-\dot{\beta}^a(t) - \ddot{\beta}^a(t) = 0. \end{aligned}$$

Normally one would continue with the \dot{q}^a terms; however, the purely time-based terms, or the terms that do not depend on \dot{q}^a or q^a , are quite simple to solve. The equation to solve is

$$\dot{\beta}^a(t) + \ddot{\beta}^a(t) = 0,$$

which gives

$$\beta^a(t) = \beta_1^a + \beta_2^a \exp(-t),$$

where β_1^a and β_2^a are constants. Thus,

$$\eta^a = \beta_1^a + \beta_2^a \exp(-t) + \gamma_b^a(t)q^b.$$

The symmetry condition equations are now

$$\begin{aligned} & -\dot{q}^a \dot{\alpha}^0(t) + \dot{q}^a \ddot{\alpha}^0(t) - 2\dot{q}^b \dot{\gamma}_b^a(t) - 2\dot{q}^b g^a \alpha^b \exp(t) - \dot{q}^a g^b \alpha^b \exp(t) \\ & - g_{,b}^a (\beta_1^b + \beta_2^b \exp(-t)) - g_{,b}^a \gamma_c^b(t) q^c - \dot{\gamma}_b^a(t) q^b - 2g^a \dot{\alpha}^0(t) \\ & - 2g^a \alpha^b \exp(t) q^b + g^b \gamma_b^a(t) - \ddot{\gamma}_b^a(t) q^b = 0. \end{aligned}$$

At this point the formation force needs to be substituted in for g^a to solve the \dot{q}^a and q^a equations. A Mathematica program was used to expedite the solving process and can be found in the Appendix.

3.3 The symmetries for the system of interest

From running the program for multiple cases, a general result emerged, regardless of the number of agents. Note that it is possible to check this result by computing XH and checking if it equals zero. This will be shown after presenting the general form of the symmetries for the system. The general solution for N agents with an

undirected connected neighbor graph is

$$\begin{aligned}
X &= \xi \frac{\partial}{\partial t} + \eta^a \frac{\partial}{\partial q^a} \\
\xi &= \alpha^0 \\
\bar{\eta} &= (\eta^1, \dots, \eta^{2N})^T = \overline{\beta(t)} + \bar{\Gamma} \bar{q}, \\
\bar{q} &= (q^1, \dots, q^{2N})^T, \quad \overline{\beta(t)} = (\beta^1(t), \dots, \beta^{2N}(t))^T, \\
\beta^i(t) &= \begin{cases} \beta_1^1 + \beta_2^1 \exp(-t) & \text{for } i \text{ odd} \\ \beta_1^2 + \beta_2^2 \exp(-t) & \text{for } i \text{ even} \end{cases}, \\
\bar{\Gamma} &= \begin{pmatrix} B & C & \dots & C \\ C & B & & \vdots \\ \vdots & & \ddots & C \\ C & \dots & C & B \end{pmatrix}, \\
B &= \begin{pmatrix} \gamma_1^1 & \gamma_2^1 \\ \gamma_1^2 & \gamma_2^2 \end{pmatrix}, \quad C = \begin{pmatrix} \gamma_1^1 & \gamma_4^1 \\ \gamma_2^1 - \gamma_4^1 + \gamma_1^2 & \gamma_2^2 \end{pmatrix}.
\end{aligned}$$

Each constant, or free variable, corresponds to a symmetry. The free variables are $\alpha^0, \beta_1^1, \beta_2^1, \beta_1^2, \beta_2^2, \gamma_1^1, \gamma_2^1, \gamma_4^1, \gamma_1^2,$ and γ_2^2 . By setting a free variable equal to one and the rest equal to zero, the individual symmetries are found. Note that the number of free variables is the same for any number of agents¹ for an undirected connected neighbor graph.

There are a total of ten free variables; therefore, it is possible to write out ten

¹When there are three agents, an additional free variable is obtained in the $\bar{\Gamma}$ matrix, which becomes a circulant matrix. This case will be shown at the end of the chapter.

unique symmetries for the system. One set of these ten symmetries is

$$\begin{aligned}
X^1 &= \frac{\partial}{\partial t} & X^2 &= \sum_{i=1}^N \frac{\partial}{\partial q^{2i-1}} & X^3 &= \sum_{i=1}^N \frac{\partial}{\partial q^{2i}} \\
X^4 &= \exp(-t) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i-1}} \right) & X^5 &= \exp(-t) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i}} \right) \\
X^6 &= \sum_{i=1}^N \left(-q^{2i} \frac{\partial}{\partial q^{2i-1}} + q^{2i-1} \frac{\partial}{\partial q^{2i}} \right) \\
X^7 &= \left(\sum_{i=1}^N q^{2i} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i-1}} \right) & X^8 &= \left(\sum_{i=1}^N q^{2i-1} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i}} \right) \\
X^9 &= \left(\sum_{i=1}^N q^{2i-1} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i-1}} \right) & X^{10} &= \left(\sum_{i=1}^N q^{2i} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i}} \right),
\end{aligned}$$

where

$$\begin{aligned}
X &= \alpha^0 X^1 + \beta_1^1 X^2 + \beta_1^2 X^3 + \beta_2^1 X^4 + \beta_2^2 X^5 + (\gamma_4^1 - \gamma_2^1) X^6 + \gamma_4^1 X^7 \\
&\quad + (\gamma_2^1 - \gamma_4^1 + \gamma_1^2) X^8 + \gamma_1^1 X^9 + \gamma_2^2 X^{10}.
\end{aligned}$$

The purpose of splitting up the general symmetry into individual symmetries is two-fold. The first is that it allows for easier computation of $XH = 0$, as each of the individual symmetries must also satisfy the relationship, or alternatively $X^i H = 0$. This is due to the definition of a symmetry being $X^i H = 0$, so if this does not hold, then X^i is not a symmetry for the system. Since X is a linear combination of the individual symmetries X^i , if $X^i H = 0$, then $XH = 0$. The second purpose is for coordinate reduction, which will be shown in Chapter 4.

These individual symmetries may not seem easier to deal with; however, they mimic the original structure of the system. Recall that the system of interest deals with agents that have a control law in the x - and y -directions. The structure of the q -dynamics is such that the x -dynamics have an odd index, while the y -dynamics have an even index. Therefore, when a summation occurs over all of the odd indices,

it corresponds to a symmetry with the x -direction. For instance, X^2 corresponds to the symmetry of a constant shift of the system in the x -direction. Likewise, X^3 , which has a summation over all of the even indices, corresponds to a constant shift in the y -direction. The other main symmetry is X^6 , which is the symmetry with respect to planar rotation.

3.4 Verification of the symmetries

Since the general solution for the symmetry, and resulting individual symmetries, was not computed by hand, the solution should be validated. This can be done quite easily by checking if $XH = 0$, as that is the definition of a symmetry for a system H . To assist in calculating $XH = 0$, the system of interest will be rewritten as two systems, or back into x and y notation essentially. Recall that $H^i = \ddot{q}^i - \omega^i = 0$. The system of interest is then

$$\begin{aligned} H^{2i-1} &= \ddot{q}^{2i-1} + \dot{q}^{2i-1} + \sum_{j \in \tilde{\mathcal{N}}_i} d_j^i (q^{2i-1} - q^l) \\ H^{2i} &= \ddot{q}^{2i} + \dot{q}^{2i} + \sum_{j \in \tilde{\mathcal{N}}_i} d_j^i (q^{2i} - q^m) \\ d_j^i &= (q^{2i-1} - q^l)^2 + (q^{2i} - q^m)^2 - \hat{d}_j^i, \end{aligned}$$

where $l = 2i - 1 + 2j \pmod{2N}$, $m = 2i + 2j \pmod{2N}$.

Now it will be rather straightforward to show that $X^i H = 0$, which shows that $XH = 0$. Recall that in order for the symmetry to act on a differential equation, it must be prolonged to include derivative terms. Additionally, since the individual symmetries two through ten do not contain an ξ term, the prolongation simplifies to $X^i = \eta^j \frac{\partial}{\partial q^j} + \dot{\eta}^j \frac{\partial}{\partial \dot{q}^j} + \ddot{\eta}^j \frac{\partial}{\partial \ddot{q}^j}$, where $\dot{\eta}^j$ and $\ddot{\eta}^j$ are exactly the first and second derivatives of η with respect to time.

1. $X^1 = \frac{\partial}{\partial t}$ Note that H^{2i-1} and H^{2i} have no standalone t terms. Therefore,

$$X^1 H = 0.$$

2. $X^2 = \sum_{i=1}^N \frac{\partial}{\partial q^{2i-1}}$ This is also the prolonged version since $\eta^{2i-1} = 1$. Note that for both H^{2i-1} and H^{2i} , any q^{2i-1} term is subtracted from q^l , where $l = 2i - 1 + 2j \pmod{2N}$. Since q^l is also contained in the set of q^{2i-1} terms, all terms will cancel out, resulting in $X^2 H = 0$.
3. $X^3 = \sum_{i=1}^N \frac{\partial}{\partial q^{2i}}$ The same logic that was used for showing that $X^2 H = 0$ holds for showing that $X^3 H = 0$.
4. $X^4 = \exp(-t) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i-1}} \right) - \exp(-t) \left(\sum_{i=1}^N \frac{\partial}{\partial \dot{q}^{2i-1}} \right) + \exp(-t) \left(\sum_{i=1}^N \frac{\partial}{\partial \ddot{q}^{2i-1}} \right)$ Since H^{2i} has no \dot{q}^{2i-1} and \ddot{q}^{2i-1} terms, $X^4 H^{2i} = 0$ by the same logic as was used for X^2 . For H^{2i-1} , the q^{2i-1} terms cancel out as in X^2 . Since there are alternating signs on the prolonged terms of X^4 and the \dot{q}^{2i-1} terms are always added to a \ddot{q}^{2i-1} term, the remaining terms cancel out as well, showing that $X^4 H^{2i-1} = 0$ and thus, $X^4 H = 0$.
5. $X^5 = \exp(-t) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i}} \right) - \exp(-t) \left(\sum_{i=1}^N \frac{\partial}{\partial \dot{q}^{2i}} \right) + \exp(-t) \left(\sum_{i=1}^N \frac{\partial}{\partial \ddot{q}^{2i}} \right)$ Again, the same logic that was used for X^4 can be used to show that $X^5 H = 0$.
6. $X^6 = \sum_{i=1}^N \left(-q^{2i} \frac{\partial}{\partial q^{2i-1}} + q^{2i-1} \frac{\partial}{\partial q^{2i}} \right) + \sum_{i=1}^N \left(-\dot{q}^{2i} \frac{\partial}{\partial \dot{q}^{2i-1}} + \dot{q}^{2i-1} \frac{\partial}{\partial \dot{q}^{2i}} \right) + \sum_{i=1}^N \left(-\ddot{q}^{2i} \frac{\partial}{\partial \ddot{q}^{2i-1}} + \ddot{q}^{2i-1} \frac{\partial}{\partial \ddot{q}^{2i}} \right)$ This is the symmetry with respect to planar rotation. Note that the distance metric d_j^i is invariant with respect to planar rotation and can be easily checked by $X^6 d_j^i = 0$. Note also that this symmetry essentially swaps the x and y terms. This results in, for a given i , $X^6 H^{2i-1} = -H^{2i}$ and $X^6 H^{2i} = H^{2i-1}$, which both equal zero since H equals zero. Therefore $X^6 H = 0$.
7. $X^7 = \left(\sum_{i=1}^N q^{2i} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i-1}} \right) + \left(\sum_{i=1}^N \dot{q}^{2i} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial \dot{q}^{2i-1}} \right) + \left(\sum_{i=1}^N \ddot{q}^{2i} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial \ddot{q}^{2i-1}} \right)$ Again, the q^{2i-1} and q^{2i} terms cancel out since there is a summation of all of the q^{2i-1} partials. The prolonged terms result in a summation of all of the \dot{q}^{2i} and \ddot{q}^{2i} terms. Note that $\sum_{i=1}^N H^{2i} = \sum_{i=1}^N (\dot{q}^{2i} + \ddot{q}^{2i})$, since all of the formation-based terms cancel out. Therefore, $X^7 H^{2i-1} = \sum_{j=1}^N H^{2j}$, which equals zero since $H = 0$, and $X^7 H^{2i} = 0$, which results in $X^7 H = 0$.
8. $X^8 = \left(\sum_{i=1}^N q^{2i-1} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i}} \right) + \left(\sum_{i=1}^N \dot{q}^{2i-1} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial \dot{q}^{2i}} \right) + \left(\sum_{i=1}^N \ddot{q}^{2i-1} \right) \left(\sum_{i=1}^N \frac{\partial}{\partial \ddot{q}^{2i}} \right)$ The same logic as for X^7 results in $X^8 H^{2i-1} = 0$ and $X^8 H^{2i} = \sum_{j=1}^N H^{2j-1} = 0$ and thus, $X^8 H = 0$.

9. $X^9 = \left(\sum_{i=1}^N q^{2i-1}\right) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i-1}}\right) + \left(\sum_{i=1}^N \dot{q}^{2i-1}\right) \left(\sum_{i=1}^N \frac{\partial}{\partial \dot{q}^{2i-1}}\right) + \left(\sum_{i=1}^N \ddot{q}^{2i-1}\right) \left(\sum_{i=1}^N \frac{\partial}{\partial \ddot{q}^{2i-1}}\right)$
The same logic as for X^7 results in $X^9 H^{2i-1} = \sum_{j=1}^N H^{2j-1} = 0$ and $X^9 H^{2i} = 0$
and thus, $X^9 H = 0$.

10. $X^{10} = \left(\sum_{i=1}^N q^{2i}\right) \left(\sum_{i=1}^N \frac{\partial}{\partial q^{2i}}\right) + \left(\sum_{i=1}^N \dot{q}^{2i}\right) \left(\sum_{i=1}^N \frac{\partial}{\partial \dot{q}^{2i}}\right) + \left(\sum_{i=1}^N \ddot{q}^{2i}\right) \left(\sum_{i=1}^N \frac{\partial}{\partial \ddot{q}^{2i}}\right)$
 $\left(\sum_{i=1}^N \frac{\partial}{\partial \ddot{q}^{2i}}\right)$ The same logic as for X^7 results in $X^{10} H^{2i-1} = 0$ and $X^{10} H^{2i} = \sum_{j=1}^N H^{2j} = 0$ and thus, $X^{10} H = 0$.

Therefore, since all of the individual symmetries result in $X^i H = 0$ and X is a linear combination of all of the individual symmetries, $XH = 0$.

This means that the general form taken from the computer program is indeed a symmetry for any number of agents with an undirected connected neighbor graph. Note that the desired distance between agents was never specified. The computer program does assume that $d_k^i = d_k^j$; however, the check of $XH = 0$ uses the general form for the system, which does not have this constraint. Therefore, since $XH = 0$ for the general form of the system, the desired distances can be anything. Additionally, the analysis does not take into account the stability or feasibility of the formation. In other words, it is possible to define desired distances such that a formation is never possible.

If a directed neighbor graph is used instead of an undirected neighbor graph, then the individual symmetries $X^1 - X^6$ still hold. However, the individual symmetries $X^7 - X^{10}$ require that the neighbor graph is connected and are therefore not symmetries for the system. This is due to $\sum_{i=1}^N H^{2i} \neq \sum_{i=1}^N (\dot{q}^{2i} + \ddot{q}^{2i})$ and $\sum_{i=1}^N H^{2i-1} \neq \sum_{i=1}^N (\dot{q}^{2i-1} + \ddot{q}^{2i-1})$. It should be noted that $X^1 - X^6$ might not be the only symmetries for a system with a directed neighbor graph, as a directed neighbor graph may lead to symmetries that an undirected neighbor graph does not have. If a disconnected neighbor graph is used, then the resulting $\bar{\Gamma}$ matrix has a block diagonal structure, where the blocks are the $\bar{\Gamma}$'s for the smaller connected components of the disconnected neighbor graph. In other words, each connected component of

the disconnected neighbor graph can be inspected individually and then combined together for the final result.

3.5 Special case for three agents

For completeness, there is an additional individual symmetry when there are three agents. This symmetry is due to the ability of the $\bar{\Gamma}$ matrix to become a circulant matrix. The $\bar{\Gamma}$ matrix for three agents is

$$\bar{\Gamma} = \begin{pmatrix} B & C & D \\ D & B & C \\ C & D & B \end{pmatrix}, \quad B = \begin{pmatrix} \gamma_1^1 & \gamma_2^1 \\ \gamma_1^2 & \gamma_2^2 \end{pmatrix},$$

$$C = \begin{pmatrix} \gamma_3^1 & \gamma_4^1 \\ \gamma_2^1 - \gamma_4^1 + \gamma_1^2 & -\gamma_1^1 + \gamma_3^1 + \gamma_2^2 \end{pmatrix}, \quad D = \begin{pmatrix} 2\gamma_1^1 - \gamma_3^1 & \gamma_4^1 \\ \gamma_2^1 - \gamma_4^1 + \gamma_1^2 & \gamma_1^1 - \gamma_3^1 + \gamma_2^2 \end{pmatrix}.$$

Note that when $\gamma_1^1 = \gamma_3^1$, matrices C and D are equal to each other and are also equal to the original C matrix in the general case.

CHAPTER 4

USING THE SYMMETRIES FOR REDUCTION

4.1 General theory and notation

Given a symmetry $X^k = \xi_k \frac{\partial}{\partial t} + \eta_k^i \frac{\partial}{\partial q^i}$, it is possible to define a coordinate transformation such that $X^k = \frac{\partial}{\partial s}$. It is through this coordinate transformation that an integrating factor is found that reduces the order of the system.

This coordinate transformation, (r^j, s) , is defined such that $X^k s = 1$ and $X^k r^j = 0$. This can be done through an educated guess of the coordinates, or through the trajectories (orbits) of the group generated by X^k .

The trajectories are

$$\frac{dq^i}{d\lambda} = \eta_k^i, \quad \frac{dt}{d\lambda} = \xi_k,$$

which can be rewritten as

$$\frac{dq^1}{\eta_k^1} = \frac{dq^2}{\eta_k^2} = \dots = \frac{dq^n}{\eta_k^n} = \frac{dt}{\xi_k}.$$

It is possible to take the initial values for these trajectories as r^j and solve $X^k s = 1$ by a line integral.

For clarity, this process is rewritten for a system with only one equation. To solve $Xr = 0$, one takes an orbit for which r equals a constant. This gives the expression

$$dr = 0 = r_{,t}dt + r_{,q}dq.$$

Note that $Xr = 0$ can be rewritten as

$$Xr = \xi r_{,t} + \eta r_{,q} = 0.$$

It is then possible to eliminate $r_{,t}$ and $r_{,q}$ by the prior two equations and obtain the expression

$$\xi dq - \eta dt = 0.$$

Now r is the constant of integration that appears in the solution to the equation above. Recall that for an orbit, r is defined as a constant. Each orbit of the system is defined by an r value, and it is possible to invert the system to obtain an expression for r in terms of q and t .

4.2 Two-agent example

For simplicity, the reduction process will be shown for the two-agent system. Additionally, since the process requires multiple coordinate transformations, a subscript will be added to indicate the step in the process. The two-agent system is

$$\begin{aligned}\ddot{q}_0^1 &= -\dot{q}_0^1 - d_1(q_0^1 - q_0^3) \\ \ddot{q}_0^2 &= -\dot{q}_0^2 - d_1(q_0^2 - q_0^4) \\ \ddot{q}_0^3 &= -\dot{q}_0^3 + d_1(q_0^1 - q_0^3) \\ \ddot{q}_0^4 &= -\dot{q}_0^4 + d_1(q_0^2 - q_0^4) \\ d_1 &= (q_0^1 - q_0^3)^2 + (q_0^2 - q_0^4)^2 - \hat{d}_1.\end{aligned}$$

For the following analysis, the symmetry X_0^1 will be ignored. This is due to it being the only individual symmetry with a partial with respect to time, and it is already in the form of $\frac{\partial}{\partial s}$.

4.2.1 Translation in x -direction symmetry

The first symmetry that will be used is $X_0^2 = \frac{\partial}{\partial q_0^1} + \frac{\partial}{\partial q_0^3}$. The method requires defining a coordinate transformation ϕ_1 such that $X_0^2 r^j = 0$ and $X_0^2 s^1 = 1$. One such transformation is

$$\phi_1(q_0^1, q_0^2, q_0^3, q_0^4) = \begin{bmatrix} q_0^1 \\ q_0^1 - q_0^3 \\ q_0^2 \\ q_0^4 \end{bmatrix} = \begin{bmatrix} s^1 \\ r^1 \\ r^2 \\ r^3 \end{bmatrix}.$$

Since this process will be repeated, the coordinates (r^1, r^2, r^3) will be changed to (q_1^1, q_1^2, q_1^3) , respectively.

The inverse transformation written in the new coordinates is

$$\phi_1^{-1}(s^1, q_1^1, q_1^2, q_1^3) = \begin{bmatrix} s^1 \\ q_1^2 \\ s^1 - q_1^1 \\ q_1^3 \end{bmatrix} = \begin{bmatrix} q_0^1 \\ q_0^2 \\ q_0^3 \\ q_0^4 \end{bmatrix}.$$

The symmetries will now be recalculated based on the new coordinate system. For brevity, these symmetries will be referred to as being the updated symmetries for the updated system. The updated symmetries are found by transforming the system to the new coordinates and recalculating the symmetries. The updated system of

equations is

$$\begin{aligned}
\ddot{s}^1 &= -\dot{s}^1 - d_1 q_1^1 \\
\ddot{q}_1^1 &= -\dot{q}_1^1 - 2d_1 q_1^1 \\
\ddot{q}_1^2 &= -\dot{q}_1^2 - d_1 (q_1^2 - q_1^3) \\
\ddot{q}_1^3 &= -\dot{q}_1^3 + d_1 (q_1^2 - q_1^3) \\
d_1 &= (q_1^1)^2 + (q_1^2 - q_1^3)^2 - \hat{d}_1.
\end{aligned}$$

Note that s^1 does not appear in the q -dynamics. The q -dynamics are the reduced system, and the s^1 equation only needs to be solved if one wants to recover the full dynamics of the system. In other words, if one were to solve the reduced system, then the s^1 equation only needs to be solved to determine where the formation is on the x -axis and its velocity in the x -direction.

The updated symmetries for the reduced q -dynamics are then

$$\begin{aligned}
X_1^1 &= \frac{\partial}{\partial t} \\
X_1^2 &= 0 & X_1^3 &= \frac{\partial}{\partial q_1^2} + \frac{\partial}{\partial q_1^3} \\
X_1^4 &= 0 & X_1^5 &= \exp(-t) \left(\frac{\partial}{\partial q_1^2} + \frac{\partial}{\partial q_1^3} \right) \\
X_1^6 &= -(q_1^2 - q_1^3) \frac{\partial}{\partial q_1^1} + \frac{1}{2} q_1^1 \left(\frac{\partial}{\partial q_1^2} - \frac{\partial}{\partial q_1^3} \right) \\
X_1^7 &= 0 & X_1^8 &= 0 \\
X_1^9 &= 0 & X_1^{10} &= (q_1^2 + q_1^3) \left(\frac{\partial}{\partial q_1^2} + \frac{\partial}{\partial q_1^3} \right).
\end{aligned}$$

4.2.2 Translation in y -direction symmetry

Now the process will be repeated with the symmetry $X_1^3 = \frac{\partial}{\partial q_1^2} + \frac{\partial}{\partial q_1^3}$. A second coordinate transformation ϕ_2 is defined such that $X_1^3 r^j = 0$ and $X_1^3 s^2 = 1$. Note that

this is s^2 as s^1 was defined in the prior section. One such transformation is

$$\phi_2(q_1^1, q_1^2, q_1^3) = \begin{bmatrix} q_1^2 \\ q_1^1 \\ q_1^2 - q_1^3 \end{bmatrix} = \begin{bmatrix} s^2 \\ r^1 \\ r^2 \end{bmatrix}.$$

Again, the coordinates (r^1, r^2) will have a change in notation to (q_2^1, q_2^2) , respectively. This gives an inverse coordinate transformation of

$$\phi_2^{-1}(s^2, q_2^1, q_2^2) = \begin{bmatrix} q_2^1 \\ s^2 \\ s^2 - q_2^2 \end{bmatrix} = \begin{bmatrix} q_1^1 \\ q_1^2 \\ q_1^3 \end{bmatrix}.$$

The transformation ϕ_2^{-1} is used to write the system in the new coordinates.

$$\begin{aligned} \ddot{s}^1 &= -\dot{s}^1 - d_1 q_2^1 \\ \ddot{s}^2 &= -\dot{s}^2 - d_1 q_2^2 \\ \ddot{q}_2^1 &= -\dot{q}_2^1 - 2d_1 q_2^1 \\ \ddot{q}_2^2 &= -\dot{q}_2^2 - 2d_1 q_2^2 \\ d_1 &= (q_2^1)^2 + (q_2^2)^2 - \hat{d}_1. \end{aligned}$$

Recall that the s^1 equation only needs to be solved if one wishes to know where on the x -axis the formation is and with what velocity the formation is traveling in the x -direction. Similarly, the s^2 equation only needs to be solved for the formation's location on the y -axis and the velocity of the formation in the y -direction. The reduced system is now only two second-order differential equations in terms of q_2^1 and q_2^2 .

The updated symmetries for the reduced q -dynamics are then

$$\begin{aligned}
X_2^1 &= \frac{\partial}{\partial t} \\
X_2^2 &= 0 & X_2^3 &= 0 \\
X_2^4 &= 0 & X_2^5 &= 0 \\
X_2^6 &= -q_2^2 \frac{\partial}{\partial q_2^1} + q_2^1 \frac{\partial}{\partial q_2^2} \\
X_2^7 &= 0 & X_2^8 &= 0 \\
X_2^9 &= 0 & X_2^{10} &= 0.
\end{aligned}$$

4.2.3 A comment on the coordinate transformations thus far

Recall that the main objective is to reduce the system for ease of further calculations, including stability of the formation, where stability is defined as the agents converging to a formation. For some applications, it may be important that the agents are at rest (Case A), while other applications are only concerned that the agents are not moving relative to each other (Case B). This method is able to handle both cases. For Case B, each coordinate transformation thus far has been able to eliminate two variables, s^i and \dot{s}^i . This is shown by the q dynamics not being dependent on s^i and \dot{s}^i . As will be shown with the rotational case, it is not always possible to eliminate two variables with one coordinate transformation; however, it is a fortuitous result when it does occur. The reduced dynamics for this case are the q -dynamics

$$\begin{aligned}
\ddot{q}_2^1 &= -\dot{q}_2^1 - 2d_1 q_2^1 \\
\ddot{q}_2^2 &= -\dot{q}_2^2 - 2d_1 q_2^2 \\
d_1 &= (q_2^1)^2 + (q_2^2)^2 - \hat{d}_1.
\end{aligned}$$

For Case A, each coordinate transformation has eliminated only one variable, which is all that the method is designed to do [18]. In this case it is convenient to define a new variable p^i such that

$$p^i = \dot{s}^i,$$

which defines the quadrature

$$s^i = \int_0^\infty p^i dt.$$

The quadrature is now the equation to be solved if one wishes to determine where on the x - and y -axis the formation is. The reduced dynamics is a system of two first order differential equations and two second order differential equations

$$\begin{aligned}\dot{p}^1 &= -p^1 - d_1 q_2^1 \\ \dot{p}^2 &= -p^2 - d_1 q_2^2 \\ \ddot{q}_2^1 &= -\dot{q}_2^1 - 2d_1 q_2^1 \\ \ddot{q}_2^2 &= -\dot{q}_2^2 - 2d_1 q_2^2 \\ d_1 &= (q_2^1)^2 + (q_2^2)^2 - \hat{d}_1.\end{aligned}$$

Since the q dynamics do not depend on the p variables, it is possible to solve the q dynamics first and then solve the p dynamics. In terms of showing stability, one would proceed the same way as with Case B to show stability of the q dynamics. Then, after that is shown, one would treat q as an input to the p dynamics and show stability of the p dynamics.

4.2.4 Rotational symmetry

Now the rotational symmetry X_2^6 will be used. For the prior two symmetries, it was convenient to keep the system as a system of second order differential equations and define the coordinate transformation in terms of only the positions. It is also

possible to define the coordinate transformation in terms of the velocities, or a mix of the position and velocity terms. Note that the prolongation of the symmetry needs to be used when including velocity terms in the coordinate transformation.

For both the original x - and y - translation symmetries, the prolongation is equal to the original symmetry. This is due to the coefficient in front of the partials being a constant. The rotational symmetry prolongates to

$$X_2^6 = -q_2^2 \frac{\partial}{\partial q_2^1} + q_2^1 \frac{\partial}{\partial q_2^2} - \dot{q}_2^2 \frac{\partial}{\partial \dot{q}_2^1} + \dot{q}_2^1 \frac{\partial}{\partial \dot{q}_2^2}.$$

To create a simplified set of reduced dynamics, a coordinate transformation will be defined such that the reduced dynamics are all first order differential equations. One such transformation is

$$\phi_3(q_2^1, q_2^2, \dot{q}_2^1, \dot{q}_2^2) = \begin{bmatrix} \arctan 2 \left(\frac{q_2^2}{q_2^1} \right) \\ (q_2^1)^2 + (q_2^2)^2 - \hat{d}_1 \\ \dot{q}_2^1 q_2^1 + \dot{q}_2^2 q_2^2 \\ ((\dot{q}_2^1)^2 + (\dot{q}_2^2)^2 + ((q_2^1)^2 + (q_2^2)^2 - \hat{d}_1)^2) \end{bmatrix} = \begin{bmatrix} s^3 \\ r^1 \\ r^2 \\ r^3 \end{bmatrix}.$$

Note that s^3 is defined in a similar manner as the argument, or angle, of a complex number expressed in exponential notation. As such, the same extended arctan function, often expressed as $\arctan 2$, will be used, where the inverse transformation is

$$\phi_3^{-1}(s^3, r^1, r^2, r^3) = \begin{bmatrix} \sqrt{r^1 + \hat{d}_1} \cos s^3 \\ \sqrt{r^1 + \hat{d}_1} \sin s^3 \end{bmatrix} = \begin{bmatrix} q_2^1 \\ q_2^2 \end{bmatrix}.$$

The transformation ϕ_3^{-1} is used to write the system in the new coordinates. For simplicity, only the q dynamics will be transformed, as the q dynamics do not depend

on the s^1 and s^2 dynamics.

$$\begin{aligned}\dot{s}^3 &= \sqrt{(r^1 + \hat{d}_1)(r^3 - (r^1)^2) - (r^2)^2} / (r^1 + \hat{d}_1) \\ \dot{r}^1 &= 2r^2 \\ \dot{r}^2 &= -2\hat{d}_1 r^1 - r^2 + r^3 - 3(r^1)^2 \\ \dot{r}^3 &= -2r^3 + 2(r^1)^2,\end{aligned}$$

where the original d_1 is equal to r^1 . Note the s^3 does not appear in any equation and the s^3 equation only needs to be solved if one wishes to know the orientation of the formation.

4.3 Five-agent example

These results will be extended to five agents. Two cases will be worked out simultaneously: Case I will have each agent only reference its nearest neighbor, while Case II will have each agent reference its nearest neighbor and its second nearest neighbor. Recall that nearest is defined with respect to the index of the agent and not with regard to spatial distance. The two cases are shown in Figure 4.1a and 4.1b, respectively.

For a general system of N agents, [14] states that a minimum of $2N - 3$ constraints are needed to define a rigid formation. Therefore, Case I does not define a rigid formation, while Case II can, provided that none of the constraints are in conflict. Two constraints are in conflict if a formation is not possible; however, removing one of the conflicting constraints will result in a formation being possible. Since it was shown that symmetries are the same regardless of what the desired distances are, the general case of the desired distances being unique will be used.

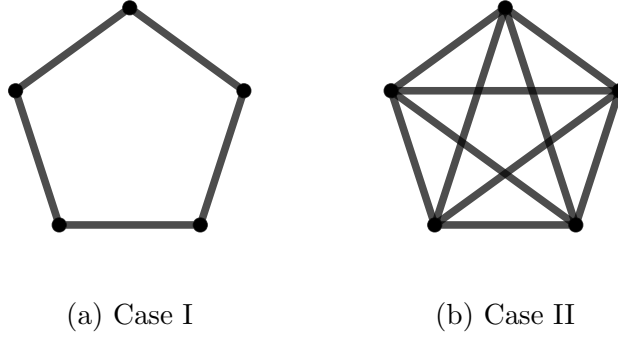


Figure 4.1. Neighbor graphs for the two five agent cases

The dynamics for Case I are

$$\ddot{q}_0^1 = -\dot{q}_0^1 - d_1^1(q_0^1 - q_0^3) - d_{-1}^1(q_0^1 - q_0^9)$$

$$\ddot{q}_0^2 = -\dot{q}_0^2 - d_1^2(q_0^2 - q_0^4) - d_{-1}^2(q_0^2 - q_0^{10})$$

$$\ddot{q}_0^3 = -\dot{q}_0^3 - d_1^3(q_0^3 - q_0^5) - d_{-1}^3(q_0^3 - q_0^1)$$

$$\ddot{q}_0^4 = -\dot{q}_0^4 - d_1^4(q_0^4 - q_0^6) - d_{-1}^4(q_0^4 - q_0^2)$$

$$\ddot{q}_0^5 = -\dot{q}_0^5 - d_1^5(q_0^5 - q_0^7) - d_{-1}^5(q_0^5 - q_0^3)$$

$$\ddot{q}_0^6 = -\dot{q}_0^6 - d_1^6(q_0^6 - q_0^8) - d_{-1}^6(q_0^6 - q_0^4)$$

$$\ddot{q}_0^7 = -\dot{q}_0^7 - d_1^7(q_0^7 - q_0^9) - d_{-1}^7(q_0^7 - q_0^5)$$

$$\ddot{q}_0^8 = -\dot{q}_0^8 - d_1^8(q_0^8 - q_0^{10}) - d_{-1}^8(q_0^8 - q_0^6)$$

$$\ddot{q}_0^9 = -\dot{q}_0^9 - d_1^9(q_0^9 - q_0^1) - d_{-1}^9(q_0^9 - q_0^7)$$

$$\ddot{q}_0^{10} = -\dot{q}_0^{10} - d_1^{10}(q_0^{10} - q_0^2) - d_{-1}^{10}(q_0^{10} - q_0^8)$$

$$d_1^1 = d_1^2 = d_{-1}^3 = d_{-1}^4 = (q_0^1 - q_0^3)^2 + (q_0^2 - q_0^4)^2 - \hat{d}_1^1$$

$$d_1^3 = d_1^4 = d_{-1}^5 = d_{-1}^6 = (q_0^3 - q_0^5)^2 + (q_0^4 - q_0^6)^2 - \hat{d}_1^3$$

$$d_1^5 = d_1^6 = d_{-1}^7 = d_{-1}^8 = (q_0^5 - q_0^7)^2 + (q_0^6 - q_0^8)^2 - \hat{d}_1^5$$

$$d_1^7 = d_1^8 = d_{-1}^9 = d_{-1}^{10} = (q_0^7 - q_0^9)^2 + (q_0^8 - q_0^{10})^2 - \hat{d}_1^7$$

$$d_1^9 = d_1^{10} = d_{-1}^1 = d_{-1}^2 = (q_0^9 - q_0^1)^2 + (q_0^{10} - q_0^2)^2 - \hat{d}_1^9.$$

The dynamics for Case II are

$$\begin{aligned}
\dot{q}_0^1 &= -\dot{q}_0^1 - d_1^1(q_0^1 - q_0^3) - d_{-1}^1(q_0^1 - q_0^9) - d_2^1(q_0^1 - q_0^5) - d_{-2}^1(q_0^1 - q_0^7) \\
\dot{q}_0^2 &= -\dot{q}_0^2 - d_1^2(q_0^2 - q_0^4) - d_{-1}^2(q_0^2 - q_0^{10}) - d_2^2(q_0^2 - q_0^6) - d_{-2}^2(q_0^2 - q_0^8) \\
\dot{q}_0^3 &= -\dot{q}_0^3 - d_1^3(q_0^3 - q_0^5) - d_{-1}^3(q_0^3 - q_0^1) - d_2^3(q_0^3 - q_0^7) - d_{-2}^3(q_0^3 - q_0^9) \\
\dot{q}_0^4 &= -\dot{q}_0^4 - d_1^4(q_0^4 - q_0^6) - d_{-1}^4(q_0^4 - q_0^2) - d_2^4(q_0^4 - q_0^8) - d_{-2}^4(q_0^4 - q_0^{10}) \\
\dot{q}_0^5 &= -\dot{q}_0^5 - d_1^5(q_0^5 - q_0^7) - d_{-1}^5(q_0^5 - q_0^3) - d_2^5(q_0^5 - q_0^9) - d_{-2}^5(q_0^5 - q_0^1) \\
\dot{q}_0^6 &= -\dot{q}_0^6 - d_1^6(q_0^6 - q_0^8) - d_{-1}^6(q_0^6 - q_0^4) - d_2^6(q_0^6 - q_0^{10}) - d_{-2}^6(q_0^6 - q_0^2) \\
\dot{q}_0^7 &= -\dot{q}_0^7 - d_1^7(q_0^7 - q_0^9) - d_{-1}^7(q_0^7 - q_0^5) - d_2^7(q_0^7 - q_0^1) - d_{-2}^7(q_0^7 - q_0^3) \\
\dot{q}_0^8 &= -\dot{q}_0^8 - d_1^8(q_0^8 - q_0^{10}) - d_{-1}^8(q_0^8 - q_0^6) - d_2^8(q_0^8 - q_0^2) - d_{-2}^8(q_0^8 - q_0^4) \\
\dot{q}_0^9 &= -\dot{q}_0^9 - d_1^9(q_0^9 - q_0^1) - d_{-1}^9(q_0^9 - q_0^7) - d_2^9(q_0^9 - q_0^3) - d_{-2}^9(q_0^9 - q_0^5) \\
\dot{q}_0^{10} &= -\dot{q}_0^{10} - d_1^{10}(q_0^{10} - q_0^2) - d_{-1}^{10}(q_0^{10} - q_0^8) - d_2^{10}(q_0^{10} - q_0^4) - d_{-2}^{10}(q_0^{10} - q_0^6) \\
d_1^1 &= d_1^2 = d_{-1}^3 = d_{-1}^4 = (q_0^1 - q_0^3)^2 + (q_0^2 - q_0^4)^2 - \hat{d}_1^1 \\
d_1^3 &= d_1^4 = d_{-1}^5 = d_{-1}^6 = (q_0^3 - q_0^5)^2 + (q_0^4 - q_0^6)^2 - \hat{d}_1^3 \\
d_1^5 &= d_1^6 = d_{-1}^7 = d_{-1}^8 = (q_0^5 - q_0^7)^2 + (q_0^6 - q_0^8)^2 - \hat{d}_1^5 \\
d_1^7 &= d_1^8 = d_{-1}^9 = d_{-1}^{10} = (q_0^7 - q_0^9)^2 + (q_0^8 - q_0^{10})^2 - \hat{d}_1^7 \\
d_1^9 &= d_1^{10} = d_{-1}^1 = d_{-1}^2 = (q_0^9 - q_0^1)^2 + (q_0^{10} - q_0^2)^2 - \hat{d}_1^9 \\
d_2^1 &= d_2^2 = d_{-2}^5 = d_{-2}^6 = (q_0^1 - q_0^5)^2 + (q_0^2 - q_0^6)^2 - \hat{d}_2^1 \\
d_2^3 &= d_2^4 = d_{-2}^7 = d_{-2}^8 = (q_0^3 - q_0^7)^2 + (q_0^4 - q_0^8)^2 - \hat{d}_2^3 \\
d_2^5 &= d_2^6 = d_{-2}^9 = d_{-2}^{10} = (q_0^5 - q_0^9)^2 + (q_0^6 - q_0^{10})^2 - \hat{d}_2^5 \\
d_2^7 &= d_2^8 = d_{-2}^1 = d_{-2}^2 = (q_0^7 - q_0^1)^2 + (q_0^8 - q_0^2)^2 - \hat{d}_2^7 \\
d_2^9 &= d_2^{10} = d_{-2}^3 = d_{-2}^4 = (q_0^9 - q_0^3)^2 + (q_0^{10} - q_0^4)^2 - \hat{d}_2^9.
\end{aligned}$$

It is not necessary to write out all of the symmetries as only three will be used

for the reduction. The first two symmetries, for both cases, are

$$X_0^2 = \frac{\partial}{\partial q_0^1} + \frac{\partial}{\partial q_0^3} + \cdots + \frac{\partial}{\partial q_0^9}$$

$$X_0^3 = \frac{\partial}{\partial q_0^2} + \frac{\partial}{\partial q_0^4} + \cdots + \frac{\partial}{\partial q_0^{10}}.$$

Note that these two symmetries correspond to the x - and y - translation, respectively. Since these symmetries are mutually exclusive and quite simple, it is possible to define a coordinate transformation that will take care of both symmetries at once.

One possible coordinate transformation, for both cases, is

$$\begin{array}{ll} s^1 = q_0^1 & s^2 = q_0^2 \\ r^1 = q_0^1 - q_0^3 & r^2 = q_0^2 - q_0^4 \\ \vdots & \vdots \\ r^7 = q_0^1 - q_0^9 & r^8 = q_0^2 - q_0^{10}. \end{array}$$

As was done with the two-agent system, r^i will be rewritten as q_1^i . The inverse coordinate transformation is

$$\begin{array}{ll} q_0^1 = s^1 & q_0^2 = s^2 \\ q_0^3 = s^1 - q_1^1 & q_0^4 = s^2 - q_1^2 \\ \vdots & \vdots \\ q_0^9 = s^1 - q_1^7 & q_0^{10} = s^2 - q_1^8. \end{array}$$

This reduces the dynamics for Case I to

$$\begin{aligned}
\dot{s}^1 &= -\dot{s}^1 - d_1^1(q_1^1) - d_1^9(q_1^7) \\
\dot{s}^2 &= -\dot{s}^2 - d_1^1(q_1^2) - d_1^9(q_1^8) \\
\ddot{q}_1^1 &= -\dot{q}_1^1 - d_1^3(q_1^3 - q_1^1) + d_1^1(q_1^1) \\
\ddot{q}_1^2 &= -\dot{q}_1^2 - d_1^3(q_1^4 - q_1^2) + d_1^1(q_1^1) \\
\ddot{q}_1^3 &= -\dot{q}_1^3 - d_1^5(q_1^5 - q_1^3) + d_1^3(q_1^3 - q_1^1) \\
\ddot{q}_1^4 &= -\dot{q}_1^4 - d_1^5(q_1^6 - q_1^4) + d_1^3(q_1^4 - q_1^2) \\
\ddot{q}_1^5 &= -\dot{q}_1^5 - d_1^7(q_1^7 - q_1^5) + d_1^5(q_1^5 - q_1^3) \\
\ddot{q}_1^6 &= -\dot{q}_1^6 - d_1^7(q_1^8 - q_1^6) + d_1^5(q_1^6 - q_1^4) \\
\ddot{q}_1^7 &= -\dot{q}_1^7 + d_1^9(q_1^7) + d_1^7(q_1^7 - q_1^5) \\
\ddot{q}_1^8 &= -\dot{q}_1^8 + d_1^9(q_1^8) + d_1^7(q_1^8 - q_1^6) \\
d_1^1 &= (q_1^1)^2 + (q_1^2)^2 - \hat{d}_1^1 \\
d_1^3 &= (q_1^3 - q_1^1)^2 + (q_1^4 - q_1^2)^2 - \hat{d}_1^3 \\
d_1^5 &= (q_1^5 - q_1^3)^2 + (q_1^6 - q_1^4)^2 - \hat{d}_1^5 \\
d_1^7 &= (q_1^7 - q_1^5)^2 + (q_1^8 - q_1^6)^2 - \hat{d}_1^7 \\
d_1^9 &= (q_1^7)^2 + (q_1^8)^2 - \hat{d}_1^9.
\end{aligned}$$

The reduced dynamics for Case II are

$$\begin{aligned}
\ddot{s}^1 &= -\dot{s}^1 - d_1^1(q_1^1) - d_1^9(q_1^7) - d_2^1(q_1^3) - d_2^7(q_1^5) \\
\ddot{s}^2 &= -\dot{s}^2 - d_1^1(q_1^2) - d_1^9(q_1^8) - d_2^1(q_1^4) - d_2^7(q_1^6) \\
\ddot{q}_1^1 &= -\dot{q}_1^1 - d_1^3(q_1^3 - q_1^1) + d_1^1(q_1^1) - d_2^3(q_1^5 - q_1^1) - d_2^9(q_1^7 - q_1^1) \\
\ddot{q}_1^2 &= -\dot{q}_1^2 - d_1^3(q_1^4 - q_1^2) + d_1^1(q_1^1) - d_2^3(q_1^6 - q_1^2) - d_2^9(q_1^8 - q_1^2) \\
\ddot{q}_1^3 &= -\dot{q}_1^3 - d_1^5(q_1^5 - q_1^3) + d_1^3(q_1^3 - q_1^1) - d_2^5(q_1^7 - q_1^3) + d_2^1(q_1^3) \\
\ddot{q}_1^4 &= -\dot{q}_1^4 - d_1^5(q_1^6 - q_1^4) + d_1^3(q_1^4 - q_1^2) - d_2^5(q_1^8 - q_1^4) + d_2^1(q_1^4) \\
\ddot{q}_1^5 &= -\dot{q}_1^5 - d_1^7(q_1^7 - q_1^5) + d_1^5(q_1^5 - q_1^3) + d_2^7(q_1^5) + d_2^3(q_1^5 - q_1^1) \\
\ddot{q}_1^6 &= -\dot{q}_1^6 - d_1^7(q_1^8 - q_1^6) + d_1^5(q_1^6 - q_1^4) + d_2^7(q_1^6) + d_2^3(q_1^6 - q_1^2) \\
\ddot{q}_1^7 &= -\dot{q}_1^7 + d_1^9(q_1^7) + d_1^7(q_1^7 - q_1^5) + d_2^9(q_1^7 - q_1^1) + d_2^5(q_1^7 - q_1^3) \\
\ddot{q}_1^8 &= -\dot{q}_1^8 + d_1^9(q_1^8) + d_1^7(q_1^8 - q_1^6) + d_2^9(q_1^8 - q_1^2) + d_2^5(q_1^8 - q_1^4) \\
d_1^1 &= (q_1^1)^2 + (q_1^2)^2 - \hat{d}_1^1 \\
d_1^3 &= (q_1^3 - q_1^1)^2 + (q_1^4 - q_1^2)^2 - \hat{d}_1^3 \\
d_1^5 &= (q_1^5 - q_1^3)^2 + (q_1^6 - q_1^4)^2 - \hat{d}_1^5 \\
d_1^7 &= (q_1^7 - q_1^5)^2 + (q_1^8 - q_1^6)^2 - \hat{d}_1^7 \\
d_1^9 &= (q_1^7)^2 + (q_1^8)^2 - \hat{d}_1^9 \\
d_2^1 &= (q_1^3)^2 + (q_1^4)^2 - \hat{d}_2^1 \\
d_2^3 &= (q_1^5 - q_1^1)^2 + (q_1^6 - q_1^2)^2 - \hat{d}_2^3 \\
d_2^5 &= (q_1^7 - q_1^3)^2 + (q_1^8 - q_1^4)^2 - \hat{d}_2^5 \\
d_2^7 &= (q_1^5)^2 + (q_1^6)^2 - \hat{d}_2^7 \\
d_2^9 &= (q_1^1 - q_1^7)^2 + (q_1^2 - q_1^8)^2 - \hat{d}_2^9.
\end{aligned}$$

Note that the reduced system defined by \ddot{s}^i and \ddot{q}_1^i does not depend on s^j , but it

does have \dot{s}^j terms. This means that it is possible to define a quadrature

$$s^i = \int_0^\infty p^i dt, \quad p^i = \dot{s}^i.$$

To continue the reduction, it is not necessary to find all of the symmetries, but rather only the one that will be used to reduce the system. As with the two-agent example, only the q -dynamics will be considered. This can be done by either the process shown in Chapter 3, or by making an educated guess and testing if $XH = 0$. The updated rotational symmetry is

$$X_1^6 = -q_1^2 \frac{\partial}{\partial q_1^1} + q_1^1 \frac{\partial}{\partial q_1^2} + \cdots - q_1^8 \frac{\partial}{\partial q_1^7} + q_1^7 \frac{\partial}{\partial q_1^8}.$$

The number of s^i and r^j terms for this coordinate transformation depends on the number of constraints that the system formation has. If there are $m \leq 2N - 3$ constraints, then $j = 1, \dots, m$ and $i = 1, \dots, 2N - m - 2$. For Case I of the five-agent case, there will be five r^j terms and three s^i terms. If there are more than $2N - 3$ constraints, then it must be checked that there are no conflicting constraints. For Case II of the five-agent case, this means that d_2^9 can be written as a function of the other d_j^i terms, or $d_2^9 = f(d_1^1, d_1^3, d_1^5, d_1^7, d_1^9, d_2^1, d_2^3, d_2^5, d_2^7)$. Since $2N - 3$ is equal to seven for five agents, it should be possible to define three d_j^i terms as a function of the remaining d_j^i terms. Note that with seven constraints, the formation is rigid; however, it can still be foldable. This means that it is possible to have a convex and a concave pentagon as possible formations. These two possibilities are shown in Figure 2.3. An additional constraint needs to be added to make it so that each agent references at least three other agents. Further details and information on this topic can be found in [1, 4, 14].

For Case I, there are five constraints. Therefore, there are three additional s^i

terms and five r^j terms. One possible set of coordinate transformations is

$$\begin{aligned}
s^3 &= \arctan 2 \left(\frac{q_1^2}{q_1^1} \right) \\
s^4 &= \arctan 2 \left(\frac{q_1^4}{q_1^3} \right) \\
s^5 &= \arctan 2 \left(\frac{q_1^6}{q_1^5} \right) \\
r^1 &= d_1^1 = (q_1^1)^2 + (q_1^2)^2 - \hat{d}_1^1 \\
r^2 &= d_1^3 = (q_1^3 - q_1^1)^2 + (q_1^4 - q_1^2)^2 - \hat{d}_1^3 \\
r^3 &= d_1^5 = (q_1^5 - q_1^3)^2 + (q_1^6 - q_1^4)^2 - \hat{d}_1^5 \\
r^4 &= d_1^7 = (q_1^7 - q_1^5)^2 + (q_1^8 - q_1^6)^2 - \hat{d}_1^7 \\
r^5 &= d_1^9 = (q_1^7)^2 + (q_1^8)^2 - \hat{d}_1^9.
\end{aligned}$$

For Case II, there are ten constraints. Therefore, only one additional s^i term is defined, and there are seven r^j terms. One possible set of coordinate transformations is

$$\begin{aligned}
s^3 &= \arctan 2 \left(\frac{q_1^2}{q_1^1} \right) \\
r^1 &= d_1^1 = (q_1^1)^2 + (q_1^2)^2 - \hat{d}_1^1 \\
r^2 &= d_1^3 = (q_1^3 - q_1^1)^2 + (q_1^4 - q_1^2)^2 - \hat{d}_1^3 \\
r^3 &= d_1^5 = (q_1^5 - q_1^3)^2 + (q_1^6 - q_1^4)^2 - \hat{d}_1^5 \\
r^4 &= d_1^7 = (q_1^7 - q_1^5)^2 + (q_1^8 - q_1^6)^2 - \hat{d}_1^7 \\
r^5 &= d_1^9 = (q_1^7)^2 + (q_1^8)^2 - \hat{d}_1^9 \\
r^6 &= d_2^1 = (q_1^3)^2 + (q_1^4)^2 - \hat{d}_2^1 \\
r^7 &= d_2^3 = (q_1^5 - q_1^1)^2 + (q_1^6 - q_1^2)^2 - \hat{d}_2^3.
\end{aligned}$$

Recall that $\arctan 2$ is the extended arctan function, where the inverse transfor-

mation is

$$\phi_3^{-1}(s^3, r^1) = \begin{bmatrix} \sqrt{r^1 + \hat{d}_1^1} \cos s^3 \\ \sqrt{r^1 + \hat{d}_1^1} \sin s^3 \end{bmatrix} = \begin{bmatrix} q_2^1 \\ q_2^2 \end{bmatrix}.$$

The advantage of defining the coordinate transformation this way is that the origin of the r^j dynamics for both cases is the set of all possible formations. Now the inverse coordinate transformation, renaming the r^j to q_2^j , and calculating the q_2^j dynamics can be done as before.

4.4 Extension of method for N agents

There are several different ways to extend these results to the general system. It is possible to pick $s^1 = q_0^1$ again, or one can use an average of all of the odd coordinates. For simplicity and ease of notation,

$$s^1 = q_0^1, \quad s^2 = q_0^2,$$

where s^1 corresponds to X_0^2 and s^2 corresponds to X_0^3 , where

$$\begin{aligned} X_0^2 &= \frac{\partial}{\partial q_0^1} + \frac{\partial}{\partial q_0^3} + \cdots + \frac{\partial}{\partial q_0^{2N-1}} \\ X_0^3 &= \frac{\partial}{\partial q_0^2} + \frac{\partial}{\partial q_0^4} + \cdots + \frac{\partial}{\partial q_0^{2N}}. \end{aligned}$$

Note that these two symmetries correspond to the x - and y - translation, respectively. Since these symmetries are mutually exclusive and quite simple, it is possible to define a coordinate transformation that will take care of both symmetries at once.

One possible coordinate transformation is

$$\begin{aligned}
s^1 &= q_0^1 & s^2 &= q_0^2 \\
r^1 &= q_0^1 - q_0^3 & r^2 &= q_0^2 - q_0^4 \\
&\vdots & &\vdots \\
r^{2N-3} &= q_0^1 - q_0^{2N-1} & r^{2N-2} &= q_0^2 - q_0^{2N}.
\end{aligned}$$

As was done with the two coordinate system, r^i will be rewritten as q_1^i . The inverse coordinate transformation is

$$\begin{aligned}
q_0^1 &= s^1 & q_0^2 &= s^2 \\
q_0^3 &= s^1 - q_1^1 & q_0^4 &= s^2 - q_1^2 \\
&\vdots & &\vdots \\
q_0^{2N-1} &= s^1 - q_1^{2N-3} & q_0^{2N} &= s^2 - q_1^{2N-2}.
\end{aligned}$$

The system of equations written in the new coordinates is now

$$\begin{aligned}
\ddot{s}^1 &= -\dot{s}^1 - \sum_{j \in \tilde{\mathcal{N}}_{s^1}} d_j^{s^1}(q_1^j) \\
\ddot{s}^2 &= -\dot{s}^2 - \sum_{j \in \tilde{\mathcal{N}}_{s^2}} d_j^{s^2}(q_1^j) \\
\ddot{q}_1^i &= -\dot{q}^i + \dot{s}^{(3+(-1)^i)/2} + \sum_{s^j \in \tilde{\mathcal{N}}_i} d_i^{s^j}(q_1^i) - \sum_{j \in \tilde{\mathcal{N}}_i} d_j^i(q_1^i - q_1^j) \\
d_j^{s^1} &= (q_1^j)^2 + (q_1^{j+1})^2 - \hat{d}_j^{s^1} \\
d_j^{s^2} &= (q_1^{j-1})^2 + (q_1^j)^2 - \hat{d}_j^{s^2} \\
d_j^i &= (q^i - q^l)^2 + (q^k - q^m)^2 - \hat{d}_j^i,
\end{aligned}$$

where $i \in \{0, \dots, 2N-2\}$, $k = i - (-1)^i$, $l = i + 2j \pmod{2N-2}$, $m = k + 2j$

(mod $2N - 2$), and care needs to be taken to ensure that $\tilde{\mathcal{N}}_i$ is defined properly to be consistent with the original system.

Note that the reduced system defined by \ddot{s}^i and \ddot{q}_1^i does not depend on s^j , but it does have \dot{s}^j terms. This means that it is possible to define a quadrature

$$s^i = \int_0^\infty p^i dt, \quad p^i = \dot{s}^i.$$

To continue the reduction, it is not necessary to find all of the symmetries, but rather only the one that will be used to reduce the system. As with the two-agent case, only the q -dynamics will be considered. The updated rotational symmetry is

$$X_1^6 = -q_1^2 \frac{\partial}{\partial q_1^1} + q_1^1 \frac{\partial}{\partial q_1^2} + \cdots - q_1^{2N-2} \frac{\partial}{\partial q_1^{2N-3}} + q_1^{2N-3} \frac{\partial}{\partial q_1^{2N-2}}.$$

From here one can choose to use $s^3 = \arctan 2 \left(\frac{q_1^2}{q_1^1} \right)$ again, or any other combination of the q^j 's that results in $X_1^6 s^i = 1$. A similar extension of the r^i 's for the five-agent coordinate transformation can be used to define the coordinate transformation for the general case. Recall that the number of s^i and r^j terms will depend on the number of constraints, which is based on the number of neighbors an agent references. If the agents that a given agent references are set arbitrarily, then it can be beneficial to construct the neighbor graph as a visual aid. Recall that in order for a formation to be rigid and not foldable, each agent must reference at least three other agents [1].

A key point to remember is that the control law determines which formations are possible, while the individual symmetries specify how these formations can move in space and still be a solution to the dynamics. Therefore, the appearance of the neighbor graph and the number of discrete reflection symmetries that it may or may not have, play no role in the number of continuous individual symmetries for the system.

It is also important to note that these coordinate transformations are found based on the symmetries. Therefore, if another second order system of equations is found to have the symmetries X_0^2 , X_0^3 , and X_0^6 , it is possible to use the same coordinate transformation to reduce the system.

CHAPTER 5

STABILITY ANALYSIS

5.1 Stability analysis for the original system

The system of interest can be expressed in x and y coordinates for each agent as

$$\begin{aligned}\ddot{x}_i &= -\dot{x}_i - \sum_{j \in \mathcal{N}_i} (x_i - x_j) d_{ij} \\ \ddot{y}_i &= -\dot{y}_i - \sum_{j \in \mathcal{N}_i} (y_i - y_j) d_{ij},\end{aligned}$$

with

$$d_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2 - \hat{d}_{ij},$$

where \hat{d}_{ij} is the square of the desired distance between agents i and j .

For this system, consider

$$V_i = \frac{1}{2}((\dot{x}_i)^2 + (\dot{y}_i)^2) + \frac{1}{8} \sum_{j \in \mathcal{N}_i} (d_{ij})^2,$$

with $V = \sum_{i=1}^N V_i$. This then gives

$$\dot{V} = - \sum_{i=1}^N ((\dot{x}_i)^2 + (\dot{y}_i)^2).$$

This is negative semi-definite, which means that it is not possible to infer asymptotic stability from Lyapunov's Theorem. One might infer notions of asymptotic stability properties from LaSalle's Principle. However, it is not straightforward to define an

invariant compact set containing all of the desired formations, as the initial conditions play a significant role in determining where the formation will be in space. In [8], this was addressed in the examples by adding a term to the control attracting the formation to the origin. This allows for the identification of an invariant compact set which can be used for LaSalle's Principle. Then, using the discrete symmetry, scaling of $\mathcal{O}(0)$ was obtained where the order is with respect to the number of agents in the system.

5.2 Lyapunov stability for the reduced two-agent system

Alternatively, one can use the reduced coordinates found in the prior chapter to show Lyapunov stability. Recall that the reduced coordinates are

$$\begin{aligned} r^1 &= (x_1 - x_2)^2 + (y_1 - y_2)^2 - \hat{d}_{12} \\ r^2 &= (x_1 - x_2)(\dot{x}_1 - \dot{x}_2) + (y_1 - y_2)(\dot{y}_1 - \dot{y}_2) \\ r^3 &= (\dot{x}_1 - \dot{x}_2)^2 + (\dot{y}_1 - \dot{y}_2)^2 + ((x_1 - x_2)^2 + (y_1 - y_2)^2 - \hat{d}_{12})^2, \end{aligned}$$

and the dynamics are given by

$$\begin{aligned} \dot{r}^1 &= 2r^2 \\ \dot{r}^2 &= -2\hat{d}_{12}r^1 - r^2 + r^3 - 3(r^1)^2 \\ \dot{r}^3 &= -2r^3 + 2(r^1)^2. \end{aligned}$$

Note that the origin of this system is the set of all possible formations. Therefore, showing stability of the origin for this system is equivalent to showing stability of the

formation. Consider the candidate Lyapunov function

$$V = \frac{1}{2} \begin{bmatrix} r^1 & r^2 \end{bmatrix} \begin{bmatrix} 2\hat{d}_{12} & \frac{1}{4} \\ \frac{1}{4} & 2 \end{bmatrix} \begin{bmatrix} r^1 \\ r^2 \end{bmatrix} + (r^3)^2,$$

which gives

$$\begin{aligned} \dot{V} &= \begin{bmatrix} r^1 & r^2 & r^3 \end{bmatrix} \begin{bmatrix} -\frac{\hat{d}_{12}}{2} & -\frac{3}{40} & \frac{1}{8} \\ -\frac{3}{40} & -\frac{3}{2} & 1 \\ \frac{1}{8} & 1 & -2 \end{bmatrix} \begin{bmatrix} r^1 \\ r^2 \\ r^3 \end{bmatrix} \\ &\quad - \frac{3}{4}(r^1)^3 + 2(r^1)^4 - 6(r^1)^2 r^2 - 2((r^1)^2 + r^3)^2. \end{aligned}$$

Using

$$\begin{bmatrix} r^1 & r^2 & r^3 \end{bmatrix} \begin{bmatrix} -\frac{\hat{d}_{12}}{2} & -\frac{3}{40} & \frac{1}{8} \\ -\frac{3}{40} & -\frac{3}{2} & 1 \\ \frac{1}{8} & 1 & -2 \end{bmatrix} \begin{bmatrix} r^1 \\ r^2 \\ r^3 \end{bmatrix} \leq \lambda_{max}((r^1)^2 + (r^2)^2 + (r^3)^2),$$

it is possible to define a domain that proves that the origin is stable, where λ_{max} is the maximum eigenvalue for the matrix. For illustrative purposes, a value of $\hat{d}_{12} = 1$ will be used; however, it is possible to show stability for values larger than 0.153 with this Lyapunov function. Note that \hat{d}_{12} must be greater than 0.03125 for the Lyapunov function to be positive definite. This then gives

$$\dot{V} \leq -0.48((r^1)^2 + (r^2)^2 + (r^3)^2) - 2((r^1)^2 + r^3)^2 - \frac{3}{4}(r^1)^3 + 2(r^1)^4 - 6(r^1)^2 r^2,$$

which is negative semi-definite for all values of r^i and negative definite for $-0.136 \leq r^1 \leq 0.172$. Recall that this shows Lyapunov stability for the origin, which is the set of all possible formations.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

In summary, the method for using symmetries to reduce a system of second order differential equations as taken from [18] was presented and applied to a multi-agent system. The multi-agent system of interest drives the agents to a desired formation in planar space. A future goal is to use the reduced dynamics to show stability of the formation, or the ability of the agents to obtain the desired formation. This is very difficult to show for the unreduced coordinates. The final orientation and location of the formation is not needed for showing formation stability. Therefore, symmetry analysis is useful in separating the orientation and location of the formation and the dynamics for the formation.

The definition of a symmetry was presented along with a way to find all of the symmetries for a system of second order differential equations. The method was applied to the system of interest, and a computer program was presented to assist in the calculation. A simplifying assumption on which neighbors an agent references was used in the computer program. The result of the computer program was checked with the definition of a symmetry and was shown to hold for the general system of interest with a connected undirected neighbor graph. Brief extensions for directed and unconnected neighbor graphs were also presented. Through the verification of the results, the general symmetry was split into individual symmetries that corresponded to actions, such as a shift in the x - and y -directions and rotation about the origin. It was shown that regardless of the number of agents in the system, the number of individual symmetries remained the same, except for the special case of three agents.

This result is independent of how many neighbors an agent references; however, an agent needs to reference at least three agents in order for a formation rigid and not foldable. The minimum number of constraints for a formation to be rigid was shown to correspond to the dimension of the reduced dynamics for a rigid formation.

The reduction process was demonstrated on the system of interest by using the individual symmetries corresponding to a constant shift in the x - and y -directions and planar rotation. Formation stability analysis was done for the two-agent case by utilizing the reduced dynamics. Two cases were presented for the five-agent case, one that was an under-constrained formation and one that was fully constrained, producing a rigid and not foldable formation. The reduction process was shown for both cases with comments on how the dimension of the reduced space corresponds to the rigidity of the formation. The results were generalized to N agents and noted that if a system of second order systems has the individual symmetries corresponding to a constant shift in the x - and y -directions and a planar rotation, then the coordinate transformation used to reduce the system would also apply to the new system.

Future work would include showing stability for the two different five-agent cases and eventually the general N agent case. Since it is possible to over-constrain the formation, it would be beneficial if stability could be shown in a way such that losing the extraneous constraints would still result in stability. The coordinate transformations presented were based on the individual symmetries. Therefore, it should be possible to define constraints on a system of second order equations such that these coordinate transformations would reduce the system. With this result it would be interesting to see if the reduction could be applied to systems that describe motion or properties not related to formations.

APPENDIX A

MATHEMATICA PROGRAM USED TO SOLVE THE SYMMETRY EQUATION

```
(* Program for computing the symmetries for the system
Subscript[x, i]''[t]==-Subscript[x, i]'[t]-Subscript[\[CapitalSigma],\
  j\[Element]Subscript[\[ScriptCapitalN], i]] \
  Subscript[d,ij](Subscript[x, i][t]-Subscript[x, j][t]);
Subscript[y, i]''[t]==-Subscript[y, i]'[t]-Subscript[\[CapitalSigma],\
  j\[Element]Subscript[\[ScriptCapitalN], i]] \
  Subscript[d, ij](Subscript[y, i][t]-Subscript[y, j][t]);
Subscript[d, ij]=(Subscript[x, i][t]-Subscript[x, j][t])^2 \
  + (Subscript[y, i][t]-Subscript[y, j][t])^2 \
  -Subscript[Overscript[d,^], ij];
where Subscript[Overscript[d,^], ij] is the desired distance between\
  agents i and j, where agent j is in the neighborhood, \
  Subscript[\[ScriptCapitalN], i], of agent i.;
```

```
-----;
Made on June 3, 2014 by Ashley Nettleman (kulczyc2@gmail.com)
with comments added on October 19, 2014.
-----;
```

```
Formula for computing the symmetries for a system of second order
differential equations is found in Hans Stephani's book
"Differential Equations: Their solution using symmetries".;
Preliminary work was done to simplify the symmetry condition
```

$\backslash[Xi]$ is a constant

$\backslash[Eta]$ is comprised of a time based term ($\backslash[Beta](t)$) and a linear combination of the position terms which is expressed as a matrix product for visual purposes.;

-----;

Inputs

NumberOfAgents: Specify the number of agents for the system

NumberOfNeighbors: Specify how far in one direction an agent will look. The program takes this input and references this number of agents to the left and right of the given agent.;

-----;

Outputs

None directly. However, it is possible to obtain any information desired by calling out these terms;

MyEtas/. gamSol//Simplify - This gives the $\backslash[Eta]$ terms;

MyGammas/.gamSol//Simplify//MatrixForm - This gives the $\backslash[CapitalGamma]$ matrix;

ControlLaws - This outputs the control laws, or the system of interest, created from the two inputs;

gamSol - This contains the constraints on the $\backslash[Gamma]$ terms that are necessary to satisfy the symmetry condition;

Note that is possible to switch from the intial assumptions to the final result by adding "/.gamSol". For instance "MyEtas" will give the intial values for the $\backslash[Eta]$ terms and "MyEtas/. gamSol" will give the final result;

-----;

Run the program by either going to Evaluation \[Rule] Evaluate Notebook or by clicking in a cell and pressing Shift+Enter; Note that there are two cells to the program. The first cell sets up the initial values and assumptions. The second cell solves the symmetry condition equations;

-----;

*)

```
ClearAll["Global`*"] (* Clears all variables *)
```

```
NumberOfAgents = 3; (* Input number of agents *)
```

```
NumberOfNeighbors = 1; (* Input for how far in each direction  
the control law will reference *)
```

```
(* Only edit the items after the line if you know what you are doing;
```

-----;

*)

```
(* The Array command is used to create a convenient way to create  
the variables and store them *)
```

```
myVar = Array[myq, 2 NumberOfAgents];
```

```
DmyVar = Array[myqd, 2 NumberOfAgents];
```

```
desiredDist = Array[mydn, NumberOfNeighbors+1, 0];
```

```
ControlLaws = Array[myw, 2 NumberOfAgents];
```

```
MyEtas = Array[eta, 2 NumberOfAgents];
```

```
MyGammas = Array[gam, {2 NumberOfAgents, 2 NumberOfAgents}];
```

```
SymmetryEqs = Array[symeq, 2 NumberOfAgents];
```

```
(* This function takes the index of an agent and calculates the
```

```

error between the actual distance and the desired distance of
the agent and an agent a given distance away. *)
mydist[qIndex_, nIndex_] = (myq[qIndex]-myq[Mod[qIndex+2 nIndex,
2 NumberofAgents,1]])^2+(myq[qIndex-(-1)^qIndex]
-myq[Mod[qIndex-(-1)^qIndex +2nIndex, 2 NumberofAgents,1]])^2
-mydn[Abs[nIndex]];

(* The control law is created based on the number of agents and
the number of neighbors an agent references. *)
For[myi=1, myi<= 2 NumberofAgents, myi++,
myw[myi] = -myqd[myi]- Sum[mydist[myi, myn] (myq[myi]
-myq[Mod[myi+2myn, 2 NumberofAgents,1]]),
{myn, -NumberofNeighbors, NumberofNeighbors}];];

(* The initial form for the \[Eta] terms *)
For[myi=1, myi<= 2 NumberofAgents, myi++,
eta[myi]=Sum[gam[myi,myj]myq[myj], {myj, 1, 2 NumberofAgents}]
+ beta[Mod[myi,2,1],1]+ beta[Mod[myi,2,1],2] Exp[-t];];

(* The symmetry condition equations with the control law and initial
form for the symmetry substituted in. *)
For[mya=1, mya<= 2 NumberofAgents, mya++,
symeq[mya]=xi D[myw[mya],t]+Sum[eta[myb] D[myw[mya], myq[myb]],
{myb, 1, 2 NumberofAgents}]+ Sum[(D[eta[myb], t]
+Sum[myqd[myc]D[eta[myb], myq[myc]],{myc, 1, 2 NumberofAgents}]
-myqd[myb]D[xi,t]-Sum[myqd[myb]myqd[myc]D[xi, myq[myc]],
{myc, 1, 2 NumberofAgents}])D[myw[mya], myqd[myb]],

```



```

{myb, 1, 2 NumberofAgents}]+2 myw[mya] (D[xi,t]
+Sum[myqd[myb] D[xi, myq[myb]],{myb, 1, 2 NumberofAgents}])
+ Sum[myw[myb] (myqd[mya] D[xi, myq[myb]]
-D[eta[mya], myq[myb]]),{myb, 1, 2 NumberofAgents}]
+Sum[myqd[mya]myqd[myb]myqd[myc]D[xi, myq[myb], myq[myc]],
{myb, 1, 2 NumberofAgents}, {myc, 1, 2 NumberofAgents}]
+ Sum[2 myqd[mya] myqd[myc] D[xi, t, myq[myc]],
{myc, 1, 2 NumberofAgents}]-Sum[myqd[myc]myqd[myb] D[eta[mya],
myq[myb], myq[myc]],{myb, 1, 2 NumberofAgents},
{myc, 1, 2 NumberofAgents}]+myqd[mya]D[xi, {t,2}]
-Sum[2 myqd[myb]D[eta[mya], t, myq[myb]],
{myb, 1, 2 NumberofAgents}]- D[eta[mya], {t,2}];];

```

```
gamSol = {}; (* Reset the value back to empty if it isn't already *)
```

(* Since the only variables left to solve are the γ terms, which are constants, it is possible to split the symmetry condition equation into multiple equations based on the powers of the position terms.;

From prior work, the highest power of the position terms will be a third order term. A convenient way to collect the coefficients of each of these terms, especially since the coefficients will be set equal to zero, is to take the partial derivative with respect to each of the three variables that comprise the third order term.;

The program goes through all of the possible third partial derivatives and stops only after it has made it through all possibilities or if the symmetry condition equation is equal to

zero.;

When the symmetry condition equation is equal to zero, there are no further constraints that can be set on the Γ terms. *)

```
For[CurrentEq=1, CurrentEq <= 2 NumberofAgents, CurrentEq++,
  loopCont = True;
  testEq =Expand[symEq[CurrentEq]/. gamSol];
  If[testEq== 0, Continue[]];
  For[myi=1, loopCont && myi<= 2 NumberofAgents, myi++,
    For[myj=1, loopCont && myj<= 2 NumberofAgents, myj++,
      For[myk=1,loopCont && myk<= 2 NumberofAgents, myk++,
        {runningSol}=Quiet[Solve[D[testEq, myq[myi], myq[myj],
          myq[myk]]==0,Flatten[MyGammas]]];
        If[runningSol== {}, Continue[]];
        gamSol=Append[gamSol/.runningSol[[1]], runningSol[[1]]];
        testEq=Expand[symEq[CurrentEq]/. gamSol];
        If[ testEq== 0,loopCont = False];]]]]]
```

BIBLIOGRAPHY

1. M. Al-Jubeh, G. Barequet, M. Ishaque, D. L. Souvaine, C. D. Tóth, and A. Winslow. Constrained tri-connected planar straight line graphs. In *Thirty Essays on Geometric Graph Theory*, pages 49–70. Springer, 2013.
2. C. Belta and V. Kumar. Abstraction and control for groups of robots. *IEEE Transactions on Robotics*, 20(5):865–875, 2004.
3. R. D’Andrea and G. E. Dullerud. Distributed control design for spatially interconnected systems. *IEEE Transactions on Automatic Control*, 48(9):1478–1495, September 2003.
4. T. Eren, W. Whiteley, and P. N. Belhumeur. A theoretical analysis of the conditions for unambiguous node localization in sensor networks. 2004.
5. B. Goodwine. Compositional stability of approximately symmetric systems: Initial results. In *Proceedings of the 21st Mediterranean Conference on Control and Automation*, pages 1470–1476. IEEE, 2013.
6. B. Goodwine. Nonlinear stability of approximately symmetric large-scale systems. In *Proceedings of the 2014 IFAC World Congress*, 2014.
7. B. Goodwine. Compositional boundedness of solutions for symmetric nonautonomous control systems. In *Proceedings of the 22nd Mediterranean Conference on Control and Automation*, 2014.
8. B. Goodwine and P. Antsaklis. Multi-agent compositional stability exploiting system symmetries. *Automatica*, 49(11):3158–3166, 2013.
9. H. Hanßmann, N. Ehrich Leonard, and T. R. Smith. Symmetry and reduction for coordinated rigid bodies. *European Journal of Control*, 12(2):176–194, 2006.
10. H. K. Khalil. *Nonlinear systems*. Prentice Hall Upper Saddle River, 3rd edition, 2002.
11. N. E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. In *Proceedings of the 40th IEEE Conference on Decision and Control*, volume 3, pages 2968–2973. IEEE, 2001.
12. M. Newman. *Networks: an introduction*. Oxford University Press, 2010.

13. R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.
14. R. Olfati-Saber and R. M. Murray. Distributed cooperative control of multiple vehicle formations using structural potential functions. In *IFAC World Congress*, pages 346–352, 2002.
15. P. J. Olver. *Applications of Lie groups to differential equations*, volume 107. Springer, 2000.
16. B. Recht and R. D’Andrea. Distributed control of systems over discrete groups. *IEEE Transactions on Automatic Control*, 49(9):1446–1452, September 2004.
17. L. Sabattini, C. Secchi, and C. Fantuzzi. Arbitrarily shaped formations of mobile robots: artificial potential fields and coordinate transformation. *Autonomous Robots*, 30(4):385–397, 2011.
18. H. Stephani. *Differential equations: their solution using symmetries*. Cambridge University Press, 1989. Malcolm MacCallum, Editor.
19. H. G. Tanner, A. Jadbabaie, and G. J. Pappas. Stable flocking of mobile agents, Part I: Fixed topology. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, volume 2, pages 2010–2015. IEEE, 2003.

<p><i>This document was prepared & typeset with pdfL^AT_EX, and formatted with NDDiss2_ε classfile (v3.2013[2013/04/16]) provided by Sameer Vijay and updated by Megan Patnott.</i></p>
